

POPULAR Computing WEEKLY

7 October 1982 Vol 1 No 25

35p

FREE COMPUTER BUY/SELL SERVICE
COMPUTER SWAP
SEE PAGE 24

Vic20 Adventure games tested

Spectrum ellipse

Boxing clever on ZX81

Machine code

Cover Story:
Laserchase on Spectrum
by Simon Lane

Win a Jupiter Ace in our Better than Basic competition

POPULAR Computing WEEKLY

The Team

Editor
Brendon Gore

Reporter
David Kelly [01-930 3271]

Sub-editor
Ninette Sharp

Editorial Secretary
Theresa Lacy

Advertisement Manager
David Lake [01-839 2846]

Advertisement Executive
Alastair Macintosh [01-930 3840]

Managing Editor
Duncan Scot

Publishing Director
Jenny Ireland

Popular Computing Weekly
Hobhouse Court, 19 Whitcomb Street
London WC2
Telephone **01-839 6835**

Published by Sunshine Publications Ltd

Typesetting, origination and printing by
Chesham Press, Chesham, Bucks

Distributed by S M Distribution
London SW9 01-274 8611 Telex 261643

Subscriptions
You can have *Popular Computing Weekly* sent to your home the subscription rate is £19.95 per year for addresses in the UK, £37.40 overseas

How to submit articles
Articles which are submitted for publication should not be more than 1000 words long

All submissions should be typed and a double space should be left between each line

Programs should whenever possible be computer printed

At present we cannot guarantee to return every submitted article so please keep a copy

Accuracy
Popular Computing Weekly cannot accept any responsibility for any errors in programs we publish although we will always try our best to make sure programs work

This Week


Stuart Hughes

Editorial

Sinclair's decision to redesign the Spectrum's printed circuit board is not altogether surprising in view of the production troubles which have plagued the Spectrum since its launch in April this year

What is surprising is Sinclair's failure to announce that it had redesigned the Spectrum pcb

A whole host of companies have grown up around the ZX80/81 and now the Spectrum They have provided hardware and software support that Sinclair either could not or would not, supply These companies have played no small part in establishing Sinclair as the No 1 micro manufacturer in the UK

Some of these companies such as DK'tronics, Downsway and East London Robotics, have produced Ram expansion boards to enable 16K Spectrum owners to upgrade their machines to 48K These expansion boards are not compatible with the redesigned pcb

It would have been politic for Sinclair to let both his customers and the rest of the industry know what was happening They found out soon enough anyway

Next Week



Can you blast your way through a meteor storm? Find out in Asteroids — a new game for ZX Spectrum.

C.P.S. GAMES

ADVENTURES — **SPECTRUM AND ATARI**

HASHA THE THIEF
Try to enter the Palace and steal the golden teapot of the Dalai Lama. There are not only traps and help is that seem some magic trying to keep you from getting to the private rooms.

THE WIZARD OF SHAM
If you can reach his hide-out, then he will give you the elixir of life. Traverse through the jungle, the ghost town of Sham and find the secret entrance to the temple in which the wizard hides. Once in the temple you will need all your wits and determination to stand the dangers awaiting you. You may meet the wizard in the end, but we doubt it.

THE FOURTH KIND
Can you manage to communicate with the extra-terrestrials and obtain from them the universal medicine for eternal life? This is not only an adventure but will and test your skills in trying to overcome what would seem to be impossible communication problems.

THE 7 CITIES OF CIBOLA
These famous cities, where the Spanish Jesuites found their gold, are situated somewhere in the South-American jungle. Their whereabouts have been lost for several centuries and nobody has found them ever since. Can you survive in the exhausting climate and find at least some treasure? And if you find it, will you still be strong enough to get home with your gold? There is not only the climate, indians, poisonous animals, secret religious sects and many more.

THE DOMED CITY
You are travelling through unmapped territory and your way is blocked by a giant ant heap. By a freak mutation these ants are as big as you and there is only one way across through the ants' lair. Some ants are friendly, others are aggressive, and your weapons are not much help, your survival depends on skill, anticipation and cunning. Will you succeed?

THE TOWER OF BRASHT
One member of your expedition has been taken prisoner by the Kharra, a cruel tribe living near the edge of civilisation. You must enlist a few companions from your team and try to get the prisoner out. Success or failure will depend on whom you choose and how they are equipped. This D&D type adventure is difficult and will take you some time to play. It can be used as a roleplaying adventure with as many players as there can be members of the team.

THE GHOST OF RAGUN
In the old haunted castle of Ragun, a large treasure is buried. Many have tried to find it, but none have ever returned to tell the tale. It is rumoured that the treasure is guarded by a ghost, who appears when least expected, and makes sure that the treasure hunter can no longer return. This adventure is definitely not for the weak hearted and we strongly advise not to play it after nightfall, especially not when you are alone in the house.

ADVENTURES FOR THE VERY YOUNG:
There is no longer any need for very young children to gaze wistfully at a computer they are not allowed to touch.
This new series of adventures is mainly based on graphics, the follow the traditional pattern of an adventure game. There are some elementary restrictions for which a bit of help from the grown ups may be needed. If you want to see some little eyes light up.

PETER RABBIT AND THE MAGIC CARROT
Peter Rabbit goes on a quest for the magic carrot. It is rumoured that any rabbit taking one bite of that carrot gets an extra twenty years of life.
Peter has to go through the big forest, meets foes (and not so foes) friends, deals with a carrot, gets help from old man oak, etc.
Will he get to the cave and find the magic carrot?

PETER RABBIT AND FATHER WILLOW
Father Willow has been damaged by vandals, and is now in a bit of a state. Peter Rabbit goes in pursuit of the vandals. They know and try not only to escape but to keep Peter Rabbit from following them. Luckily the latter gets help from the other trees, who heard about the story. But will he find the vandals and have them locked up?

PETER RABBIT AND THE NAUGHTY OWL
Jimmy the Owl has been insufferable of late. The Council of the Meadows sends Peter Rabbit on an expedition to find the Master of the Owls, in order to have Jimmy taught some manners. The Master lives very far away and it's quite an adventure getting there. Will Peter Rabbit come back without having seen the Master and thus Jimmy remain a nuisance?
It now transpires that the Peter Rabbit Adventures can be dangerously addictive to grown ups.

TUMMY DIGS
Complementing the Peter Rabbit series, a new series on Tummy Digs, a little dwarf. As with the Peter Rabbit games, the adventures are very easy (basically a maze) with graphics and it is up to the kids to invent the story themselves, after an introduction had been given.

TUMMY DIGS GOES SHOPPING
Make a shopping list, walk out of the forest and shop in town. You need find the shops, pay for your purchases and make sure that you can carry it all. Also don't run out of money.

TUMMY DIGS GOES WALKING IN THE FOREST
Have a pleasant but adventurous walk in the forest. Meet some animals and plants, have a chat, and make sure you are home in time for tea.

WAR GAMES
All with full graphics of the battle field and inclusive of manual.

KING ARTHUR
Britain in the 5th century. THE ANGLES AND SAXONS are marauding through the Country leaving behind a trail of blood and devastation. In the South a man is gathering troops and filling them for... His name is Arthur. You take his role in this fascinating wargame. Will you be there to win all the battles he won and free Britain from the plundering marauders? How good are you at commanding troops, finding the enemy and bring him to battle, sifting information, seeing through the fog of war, deploying your troops and many more similar skills?

BATTLE OF THE BULGE
Ardennes, 1944. The famous von Rundstedt offensive.

BATTLE OF THE RIVER PLATE
A simulation of this well known sea battle.

CONVOY
You are the commodore of a convoy under attack from submarines. Instant decisions are required and if you hesitate too long the damage might be worse. Try and locate the enemy and destroy him. Not easy. Again graphics but combined with verbal information.

All these games are available for ATARI 16K and SPECTRUM 16K
Some of the games will load different programs successively and are thus much larger than 16K
All C.P.S. Games, except those for children, are priced at £9.50. The Peter Rabbit and Tummy Digs games are now £4.50.

C.P.S. 14 Britton St., London EC1M 5NQ (01-251 3090)

REDDITCH ELECTRONICS
21 FERNEY HILL AVENUE
REDDITCH
WORCS B97 4RU

ZX81 HARDWARE
(Postage 40p under £4)

Keyboard kit £17.00 Built £20

In/out port Kit £16.95 Built £16.95

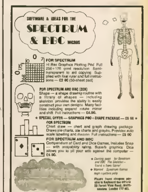
Music board Kit £16.95 Built £16.95

Female connector 23 way £2.50

Male connector 23 way £1.60

Plus books, motherboards, cables, etc.

SPECTRUM HARDWARE
Connectors etc See catalogue
Send SAE for Free Catalogue.



Spectrum plugs in new sockets

SOME memory expansion adds-on already being manufactured for the ZX Spectrum are no longer compatible with machines now being sent out

In order to solve some of the production difficulties which have plagued the Sinclair machine since its launch, the main printed-circuit board has now been redesigned The main feature is the incorporation of the Ram expansion sockets into the main board rather than as a plug-in auxiliary board

This change has caused problems for those companies who already manufacture Spectrum Ram expansion boards, since their products are no longer compatible The three companies concerned are East London Robotics, DK'tronics and Downsway

All three plan to adapt and produce two types of Ram expansion kits

Mark Vellacott of East London Robotics said that his company would continue to produce its present 32K and 64K plug-in boards for £32 and £50 However, he is now also selling both sizes of up-grade suitable for use with the new boards The new 32K version now consists of just the 12 memory chips which plug directly into the new boards and costs £25 The 64K expansion now bolts on rather than plugging in and will still cost £50

A spokeswoman for DK'tronics admitted "The new board caused a few hiccups here — because we were nt at first aware of it — but we shall just change our design "

Downsway's Martin Dore said that he was aware of the Spectrum design change "As soon as we can get hold of one of the new machines," he said, "we shall be able to produce a modified version of our Ram expansion but, like so many others, we still have machines on order Our present 32K expansion costs £42 50 and we hope the new one will be even cheaper "

Machines with the new and old boards can be distinguished, without taking them apart, by looking at the expansion ports The conducting strips on the printed-circuit visible at the port, were about the same width as the gaps between them in the original The gaps on the new pcb are much narrower than the conducting strips

More than 45 000 visitors attended the fifth Personal Computer World Show held in the Barbican Centre Exhibition Halls from September 9 to 12 See show report, page 11

Crawley IT exhibition

CRAWLEY Information Technology Exhibition takes place on October 13 to 16

It will feature more than 30 stands Admission is free On Wednesday the emphasis will be on schools and on the educational aspects of microcomputers Thursday and Friday will concentrate on business users and Saturday will be aimed at the home user and games player

On Wednesday and Saturday the exhibition will be open from 9 30 am to 5 pm, Thursday until 8 30 pm

The IT show will be held at Crawley College College Road, Crawley For further information contact Robin Green on Crawley 25686

Self-financing move by IT centres

A high-resolution graphics board for the ZX81 has been launched by the country's Information Technology Centres in a move to make them partly self-financing

The boards for use with the 16K ZX81, will cost £27 50 and will be designed and built by the IT centres at Notting Dale in London, and at Telford

It is hoped that up to 100 IT centres will be set up financed in part by £30m from the Manpower Services Commission and the Department of Industry The remainder of their running costs has to be found by the IT centres themselves

Dragon breathes fire into software market

METTOY has followed up the launch of its Dragon-32 microcomputer with a range of software cartridges and cassettes

It is now offering seven new games cartridges and a selection of games and utility cassettes The cartridges are *Berserk, Meteoroids, Cosmic Invaders, Ghost Attack, Cave Hunter, Starship Chameleon* and *Astroblast* They all cost £19 95 except *Ghost Attack* which is £24 95

The cassettes are a *Compendium of Games* a *Compendium of Applications,* five adventure games — including *Dragon Mountain, Madness and the Minotaur* and *Quest* — a *Personal Finance Package,* a *Graphic Animator,* a *Computer Voice* and a maze game, *Flag* All the cassettes cost £7 95

Andy Redman, Dragon Data s Software Development Manager, said "We are obviously looking to get into the domestic and educational software markets We think the market is as much to do with the software as the hardware — but you cannot sell the petrol until you have sold the car!

"Now that the Dragon-32 is selling so well we are stepping up the software development side of the company Not only that — when we get the disc drives for the Dragon its potential for software will be increased enormously "

Timex Challenger

Space attack held at arm's length

NOW you need never be separated from the world of the video game

Timex now offers a wrist watch, The Challenger, that plays a version of *Missile Attack*

The object of the game is to pilot your space-craft across the screen to the safety of the mothercraft, whilst avoiding the succession of enemy missiles which are trying to annihilate you at every opportunity

As well as enabling you to fight an avalanche of missiles accompanied by simulated firing noises the instrument tells the time Its liquid-crystal display also functions as a 24-hour chronograph

The Challenger is available from most Timex stockists, price £19 95

ZX81 software price cuts

IN what could be just the start of a price-cutting avalanche Quicksilva has dropped the price of its best selling ZX81 software

The cost of its *Asteroids* and *Scramble* programs has been cut from £5 50 to £4 95

Quicksilva s Mark Eyles said " We have cut the costs of the cassettes to keep the ZX81 market going There was certainly a lull in software sales after the Spectrum launch so these price drops should make it a bit more healthy "

Letters

write to Letters, Popular Computing Weekly, Hobhouse Court, 19 Whitcomb Street, London WC2

A Sassenach's error

May I take you to task I... certain errors in programs. It is interesting to put them right.

But glancing through *PCW* July 8 I saw an article entitled Up among the highlands. Good I thought but no — another clanger. Please note Edinburgh is not in the highlands.

E Miller
15 Elmgrove
Achareidh
Nairn
Scotland

Spectrum's secret passageways

I have had my Spectrum for a month and I have begun to look deeper at one or two of its secrets.

One point which may be of interest is that over 1K of the 15K Rom apparently has no thing in it. To be precise addresses 14446 to 15615 contain FF-HEX (the character set starts at 15616).

That means that 1169 bytes are unused. If that is the case, why could Sinclair not provide us with a Renumber routine, or additional commands such as *If — Then — Else* and *Repeat — Until*.

David Poole
28 Cuttys Lane
Stevenage
Hertfordshire SG1 1UN

Soft landing on Jupiter

It did not seem possible to soft-land on Jupiter in the Voyager program (*PCW* July 22). The additional procedure below allows this fairly easily, but although any key will relaunch, only one allows much chance of getting away. The additional slight changes limit mission time and add a few more velocity restraints, but why does one sometimes go into outer space when attempting to get off Jupiter — unintended, can anyone explain?

```
1640 DEF PROCsoft land
1660 UI%=0 UJ%=1
1660 VDU28 0 3 19 1 PRINT
     Welcome to Jupiter Lift off
     for Saturn" VDU28,0 31 31 26
     26
1670 XS%=XS%+70 YS%=YS%
     +70
1680 Z=GET DRAWXS% YS% B%
     =POINT(XS% YS%)
1690 UJ%=10 UI% 8:SOUND
     0 -15 7 10
1700 ENDPROC
Other changes
 105 TIME=0
 110 REPEAT
 180 PROCnew-coords
 130 UNTIL(B%<>0AND
     B%<>3)ORAB 31
     ((UI%+UJ%/10)+UJ%)
     >30ORTIME>6000
 120 IF B%=1 AND ABS(UI%+UJ%)
     <5 THEN PROCsoft-land ELSE
     140
 120 GOTO 110
 140 IFB% =1 THEN PROCmissed
     ELSE 150
 145 GOTO 100
 150 IFB%=2 THEN PROCland
     ELSE 160
 155 GOTO 100
 160 IFB%=1 THEN PROCjupiter
     ELSE 170
 166 GOTO 100
 170 IF ABS(UI%+UI%/10)+UJ%)
     >30 THEN PROCoverdrive
     ELSE 175
 172 GOTO100
 175 VDU26 PRINT TAB(5,5)
     "WHERE ARE YOU ?
 300 N=8015 (otherwise you never
     get off Jupiter)
 800 SOUND0 -15 7 10 (the motors
     go off after a time on Jupiter)
1060 SOUND17 0 (UI%+UJ%/10)
     +UJ%+10 255
1066 SOUND0 -15 7 10
1320 VDU26
Delete 1330
1264 VDU 90
Delete 1270
```

J H Powell
49 Meadowhead
Sheffield
South Yorkshire

One man's meat is another's poison

Being an ex-owner of an Atari Videopac computer system — due to the extortionate cost of its cassette games — I am dismayed at the news that Atari is taking action against software companies who are producing games similar to its own.

I am now a proud owner of a BBC model B and wish to purchase a *Pacman* game for it. But if Atari (who, on past experience, appear just to be interested in making as much profit as possible) goes through with its threats there will be no chance for me to buy this program.

Of course, I could go and stand in a queue for the machine at the pub but that is not my idea of a good night out.

PS The Saturn program was great.

Claire Hallworth
Charlston
Shay Lane
Halebarns
Altrincham
Cheshire

Speeding up screen clearing

I am 14 years old and the proud owner of a ZX81. I am part way through designing a "3D Lunar Lander" game which includes the use of the *Scroll* function. All very well, but a problem came when clearing the screen after *Scrolling*. I found out, as I know many other ZX-users have, that it can take up to 25 seconds to clear the screen after the *Scroll* function has been used. I found out that if *Poke* 16389,76 was added at line 1 of the program the problem was solved — the screen cleared instantly when told to do so. For example, try this short program.

```
Lines
1 POKE 16389 76
10 PRINT AT 21 RND*31  *
20 SCROLL
30 IF INKEY$ = 9 THEN GOTO 80
40 GOTO 10
50 CLS
```

After a few seconds of *Scrolling*, press the "9" key and watch carefully to see how fast the screen clears. Another advantage is that it speeds up character movement on-screen. For example, type in this routine.

```
Lines
1  POKE 16389 76
10 FOR F = 0 TO 30
20 PRINT AT 10 F  (space)*
30 NEXT F
```

Fast, isn't it?

One more thing, this command is not affected by *New*. Deleting it from your program will not have any effect once the program has been *run*. To erase it from your memory you will have to take the drastic step of disconnecting the lead from your computer.

I hope these hints have been of value to all you ZX81 owners.

Simon Brewer
55 Scott Avenue
Baxenden
Accrington
Lancashire BB5 2XA

An abundance of errors

I was very interested to read the letter by Ian Logan in your latest issue (*Popular Computing Weekly* September 9) regarding bugs in the Spectrum Rom. I have discovered two more — it thinks that *Int*-65535 5=-1E-38 and *Input* statements do not actually require any variables. As an example enter

```
10 INPUT "This is a bug"
20 GOTO 10
```

There are also a lot of misprints in the manual. On page 152, exercise 1, the *Sin* program mentioned is in chapter 17 not 19, and on page 170, negative numbers are represented by the number +65536, not 131072.

The program on page 176 works fine on the ZX81, but for the Spectrum line 20 should be *Print Peek (Peek* 23627+256*+Peek* 23628+n) in the first example and *Print Peek (Peek* 23635+256*Peek* 23636+n) in the second. On page 184, Chr$47 is actually "\", not "/", and on page 202 the command *Delete* "filename" is mentioned, despite the fact that it does not exist.

Both the Rom and the manual were written by Steve Vickers, so lets hope the Rom in his new Jupiter Ace has fewer bugs (even his photo on page 13 of your magazine was printed backwards).

As a final point, on Peek and Poke in the same issue, you say that the Microdrive routines are contained in the Spectrum Rom. They are not, only the facility for them is there. There is a 2K unused sector in the Rom, so this is where the routines are likely to go, in a new Rom supplied with the Microdrive, hopefully bug-free.

Andrew Pennell
14 Sweyn Road
Cliftonville
Kent

Mea Culpa. The photograph of Steve Vickers was reversed, though it takes a keen eye to spot it.

You are also correct in saying that the Microdrive routines are not contained in the Spectrum Rom, just the potential for their inclusion. We expect the Microdrive, when it finally appears, will contain its own Rom.

If you have an opinion you want to express, or have spotted an error that needs correcting, write to Letters, Popular Computing Weekly, *Hobhouse Court, 19 Whitcomb Street, London WC2*



Laser Chase

A new game for 16K Spectrum
by Simon Lane

You were designed for the Laser Chase. That is your purpose your life your future

It is 2047 You have been genetically synthesised as a player in the game And that is all All there is The game

Only earth members Stratum 1 can watch you as you win or die Other orders are prohibited But you can be sure that all those who can will be there The hologrescope auditoriums are certain to be packed

As an android you have been brilliantly designed The emotions you have been allowed are sufficient to make you want to win To not die And go forward to the next Laser Chase

All that you can concieve, all that you can ever be aware of, is contained within The Pen

Inside The Pen there is only you and your adversary When the Laser Chase begins you have been conditioned to move There is no choice Life for the Chaser Android is compressed to five decisions only four directions and a choice of speeds As you move a genetically constructed wall is cultivated in your wake — and this construction drains your energy and saps your life material

To win is to play again You win by forcing your opponent to strike either your wall or the perimeter of The Pen Both are secreted with a deadly poison which unravells your central nervous system

If you collide with the rival android's wall or the sides of The Pen you loose one of your eight lives

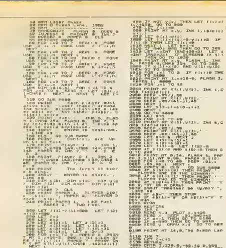
As your life banks become drained by production of the wall you may replenish from the fuel dumps Care must be taken not to restore at high speed — saturation of your genetic storage zones is fatal

Allow yourself no emotion Concentrate on the game Many incorrectly programmed Chasers have sensed freedom outside The Pen and have been ceased-out as they tried to escape

The next Laser Chase will include you if you can survive eight rounds in The Pen

Laser Chase is a game for two players on the 16K Spectrum Further instructions will be transmitted to you as the game begins

```
10 REM Laser Chase
20 REM © Simon Lane, 1982
25 REM -------------------
30 RANDOMIZE : FLASH 0: OVER 0
40 INVERSE 0: BRIGHT 0: INK 7
50 BORDER 0: PAPER 0: CLS
55 RESTORE 9000
60 FOR i=0 TO 7: READ n: POKE USR "a"+i,n: POKE USR "c"+7-i,n: NEXT i
70 FOR i=0 TO 7: READ n: POKE USR "b"+i,n: NEXT i
80 FOR i=0 TO 23: READ n: POKE USR "d"+i,n: NEXT i
90 FOR i=0 TO 7: READ n: POKE USR "g"+i,n: POKE USR "h"+7-i,n: NEXT i
100 FOR i=0 TO 7: READ n: POKE USR "j"+i,n: POKE USR "i"+7-i,n: NEXT i
102 FOR i=0 TO 7: READ n: POKE USR "k"+i,n: NEXT i
105 DIM l$(4,4): FOR i=1 TO 4: FOR j=1 TO 4: READ n: LET l$(i,j)=CHR$ n: NEXT j: NEXT i: LET c=1
110 GO SUB 6000
120 PRINT "  Each player must drive his laser chaser around the grid without hitting a laser wall or the other player. Collect extra"
130 PRINT "fuel ( ";INK 6; FLASH 1;CHR$ 156; FLASH 0; INK 7;" ) by colliding with it at slow speed to avoid explosion."
140 INPUT "  ENTER to continue."; LINE a$
150 CLS : GO SUB 6000
160 PRINT "  Controls are Up Dn Lt Rt Fast"
170 PRINT "Player 1 ( ";INK 1; PAPER 7;CHR$ 148;CHR$ 149;CHR$ 145; PAPER 0; INK 7;" ) 4 D E R 3"
180 PRINT "Player 2 ( ";INK 2; PAPER 7;CHR$ 148;CHR$ 149;CHR$ 145; PAPER 0; INK 7;" ) 0 L O P 9"
190 PRINT "  The first to score 8 wins."
200 INPUT "  ENTER to start."; LINE a$
210 DIM s(2): DIM x(2): DIM y(2): DIM f(2): DIM d(2): DIM v(2): DIM o(2)
220 PAPER 7: CLS
230 PRINT PAPER 1;" PLAYER Score ";s(1);"  "; PAPER 2;" PLAYER 2 Score ";s(2);
240 PRINT PAPER 1;" ONE Fuel "; PAPER 2;" TWO Fuel "
260 LET f(1)=f(1)+500: LET f(2)=f(2)+500
262 LET f=1
265 LET a=2
270 LET d(1)=1: LET d(2)=1
280 LET x(1)=21: LET y(1)=0
290 LET x(2)=21: LET y(2)=31
295 LET o(1)=1: LET o(2)=1
300 FOR i=1 TO 2: PAPER 8: PRINT AT 1,i*16-5; PAPER i;(" " AND f(i)<1000)+STR$ f(i);" "; PAPER 7: PRINT INK i;AT x(i),y(i);CHR$ (143+d(i)): NEXT i
310 IF f(1)=0 OR f(2)=0 OR (x(1)=x(2) AND y(1)=y(2)) THEN GO TO 2000
330 IF f THEN FOR i=1 TO 0 STEP -1: FOR j=24 TO 12 STEP -2: BEEP 0.01,i+j: NEXT j: NEXT i: FOR i=12 TO 36 STEP 2: BEEP 0.01,i: NEXT i: LET f=0
340 LET x$=("1" AND (IN 63486=247 OR IN 63486=243))+("2" AND IN 64510=247)+("3" AND IN 65022=251)+("4" AND IN 64510=251)
350 IF LEN x$=1 THEN LET d(1)=VAL x$
360 LET x$=("1" AND (IN 61438=254 OR IN 61438=252))+("2" AND IN 57342=254)+("3" AND IN 49150=253)+("4" AND IN 57342=253)
370 IF LEN x$=1 THEN LET d(2)=VAL x$
380 LET v(1)=IN 63486=253 OR IN 63486=243
390 LET v(2)=IN 61438=253 OR IN 61438=252
400 FOR i=1 TO 2: LET x=x(i): LET y=y(i): IF v(i) THEN BEEP .01,i*2+40: BEEP .01,i*2+41: GO TO 420
410 IF NOT a THEN FOR z=i*2+30 TO i*2+30 STEP 2: BEEP .01,z: NEXT z: GO TO 1000
420 LET x(i)=x(i)+(d(i)=3)-(d(i)=1): LET y(i)=y(i)+(d(i)=2)-(d(i)=4)
430 IF x(i)>21 OR x(i)<2 OR y(i)>31 OR y(i)<0 THEN LET x(i)=x: LET y(i)=y: LET f(i)=0: GO TO 900
440 IF ATTR (x(i),y(i))=57 OR ATTR (x(i),y(i))=58 THEN LET f(i)=0: GO TO 900
450 IF ATTR (x(i),y(i))<>134 THEN GO TO 900
460 IF NOT v(i) THEN LET f(i)=f(i)+250: GO TO 800
470 LET f(i)=0
800 PRINT AT x,y; INK i;l$(o(i),d(i))
810 LET o(i)=d(i)
1000 LET f(i)=f(i)-8-v(i)*8: IF f(i)<0 THEN LET f(i)=0
1010 NEXT i: LET a=1-a
1020 IF RND>.005 THEN GO TO 300
1030 LET x=INT (RND*20+2): LET y=INT (RND*32): IF SCREEN$ (x,y)<>" " THEN GO TO 1030
1040 PRINT AT x,y; FLASH 1; INK 6; PAPER 8;CHR$ 156: GO TO 300
2000 IF x(1)=x(2) AND y(1)=y(2) THEN GO TO 2500
2010 FOR i=1 TO 2: IF f(i)>0 THEN GO TO 2400
2020 PRINT AT 1,i*16-5; FLASH 1; PAPER i;"0000"
2030 FOR j=1 TO 40
2040 PRINT AT x(i),y(i); INK i;CHR$ (143+d(i))
2050 BEEP .05/j,50
2060 PRINT AT x(i),y(i);" "
2070 BEEP .05/(41-j),40
2080 NEXT j
2090 LET s(3-i)=s(3-i)+1
2400 NEXT i
2410 GO TO 3000
2500 LET a=1
2505 FOR j=1 TO 40
2510 PRINT AT x(1),y(1); INK a;CHR$ (143+d(a))
2520 BEEP .05/j,50
2530 PRINT AT x(1),y(1);" "
2540 BEEP .05/(41-j),40
2550 LET a=3-a: NEXT j
2560 LET s(1)=s(1)+1
2570 LET s(2)=s(2)+1
2580 LET f(1)=0: LET f(2)=0
3000 IF s(1)<8 AND s(2)<8 THEN GO TO 220
3010 PRINT AT 0,14; FLASH 1; PAPER 1;s(1);AT 0,30; PAPER 2;s(2)
3020 FOR i=0 TO 40: BEEP .01,i: BEEP .01,40-i: NEXT i
3030 PAPER 0: CLS : GO SUB 6000
3040 IF s(1)>s(2) THEN PRINT "  PLAYER ONE IS THE WINNER!"
3050 IF s(2)>s(1) THEN PRINT "  PLAYER TWO IS THE WINNER!"
3060 IF s(2)=s(1) THEN PRINT TAB 9;"IT IS A DRAW!"
3070 INPUT "Another go (y/n)? "; LINE a$
3080 IF a$="" THEN STOP
3090 IF a$(1)="y" OR a$(1)="Y" THEN RUN
5999 STOP
6000 RESTORE
6005 INK c
6010 READ x,y: PLOT x,y+36
6020 READ x: BEEP .01,RND*40
6030 IF x=-1 THEN GO TO 6100
6040 IF x=999 THEN GO TO 6010
6050 READ y: DRAW x,y: GO TO 6020
6100 PRINT AT 16,0;"by Simon Lane"
6110 INK 7
6120 LET c=c+1
6130 RETURN
7000 DATA 1,130,0,-48,48,0,999
7010 DATA 52,91,24,48,24,-48,999
7020 DATA 64,118,24,0,999
7030 DATA 151,136,-48,0,0,-24,48,0,0,-24,-48,0,999
7040 DATA 202,139,-48,0,0,-48,48,0,999
7050 DATA 154,118,40,0,999
7060 DATA 205,91,0,48,48,0,0,-24,-48,0,999
7070 DATA 229,118,24,-24,999
7080 DATA 49,81,-48,0,0,-48,48,0,999
7090 DATA 58,81,0,-48,999
7100 DATA 32,57,48,0,999
7110 DATA 130,81,0,-48,999
7120 DATA 103,33,24,48,24,-48,999
7130 DATA 115,57,24,0,999
7140 DATA 202,81,-48,0,0,-24,48,0,0,-24,-48,0,999
7150 DATA 253,81,-48,0,0,-48,48,0,999
7160 DATA 208,57,48,0,-1
9000 DATA 24,24,60,90,126,189,165,66
9010 DATA 0,240,124,63,63,124,240,0
9020 DATA 0,15,62,252,252,62,15,0
9030 DATA 66,60,60,90,90,66,66,60
9040 DATA 0,255,0,24,24,0,255,0
9050 DATA 60,67,64,66,66,64,127,0
9060 DATA 66,194,2,26,26,2,254,0
9070 DATA 24,60,102,195,195,102,60,24
9080 DATA 148,151,148,152
9090 DATA 153,149,152,149
9100 DATA 148,150,148,153
9110 DATA 150,149,151,149
```

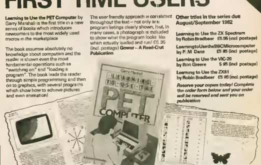


READ-OUT

READ OUT FOR SOFTWARE & BOOKS

THE FIRST IN A NEW SERIES FOR FIRST TIME USERS

JUST AVAILABLE



Learning to Use the PET Computer by Garry Marshall is the first title in a new series of books which introduces newcomers to the most widely used micros in the marketplace

The book assumes absolutely no knowledge about computers and the reader is shown even the most fundamental operations such as "switching on" and "loading a program". The book leads the reader through simple programming and then on to graphics, with several programs which show how to achieve pictures and even animation!

The user friendly approach is consistent throughout the text – not only are program listings clearly shown, but, in many cases, a photograph is included to show what the program looks like when actually loaded and run! £5.95 (incl. postage) **Gower – A Read-Out Publication**

Other titles in the series due August/September 1982

Learning to Use the ZX Spectrum by Robin Bradbeer £5.95 (incl postage)

Learning to Use the BBC Microcomputer by P. M. Dane £5.95 (incl postage)

Learning to Use the VIC-20 by Ron Greere 5.95 (incl postage)

Learning to Use the ZX81 by Robin Bradbeer £5.95 (incl. postage)

Reserve your copies today! Complete the order form below and your order will be reserved and sent you on publication

READ-OUT PUBLISHING COMPANY LTD
8 Camp Road, Farnborough, Hampshire, GU24 6EW Telephone 0252 510331/2 Telex 858001 GOWER G

READ-OUT PUBLISHING COMPANY LTD

8 CAMP ROAD, FARNBOROUGH HAMPSHIRE GU24 6EW
24 hour answering service Telephone 0252 510331/2

Name ______________________

Address ______________________

Make cheques payable to Read Out Publishing Company Ltd
I enclose my cheque for £

Please debit my Access ☐ Signed ______
Number ☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐ Date ______

Please send me copy/ies of
All prices include postage

- ☐ **Learning to Use the PET Computer @ £5.95**
- ☐ **Learning to Use the ZX Spectrum @ £5.95**
- ☐ **Learning to Use the BBC Microcomputer @ £5.95**
- ☐ **Learning to Use the VIC-20 @ £5.95**
- ☐ **Learning to Use the ZX81 @ £5.95**

PCW4

Barbican and Bingley show their wares

***David Kelly** reports on the Personal Computer World and Microscene shows*

Like a whirlwind the 5th *Personal Computer World* show has come and gone According to an independent survey conducted by the organisers over 45 500 people attended the giant exhibition spread over four days from September 9 to 12

It was one of the first events to be staged at the new purpose built Barbican Centre exhibition halls — and pretty claustrophobic it was too On the upper floor the ceiling was within easy reach and, despite the fact that the air conditioning was going full blast, the temperature soared

Still, if you could put up with the free sauna and were prepared to push your way through the crowds there was plenty to see

All the major companies were represented Some — like Atari and Commodore — displayed their range with a flare bordering on the overdone Rotating plinths and back-lit screens sell microcomputers

Sadly, however, many of the smaller home micro companies did not attend — probably becaues of cost Stands were charged at £92 per square metre, with the minimum size costing over £450

Several new machines made their first public appearances at the show The Lynx, Jupiter Ace and Colour Genie were all on display and attracted a good deal of interest It was also the first time Mettoy had exhibited the Dragon 32 and its stand drew big crowds Attention was paid to the new Commodore 64, due to go on sale in the UK towards the end of September

The show was also notable for its absentees Neither the Acorn Electron nor the Sinclair Microdrives put in an appearance

The Sinclair stand was besieged by customers waiting for their ZX Spectrums Those working on the stand spent almost more time dealing with delivery enquiries than they did selling the machine or promoting the new range of Sinclair Spectrum software They also advised waiting customers against buying non Sinclair plug-in Ram boards — the Spectrum printed-circuit board is once again being modified

Most software companies had new material — among them Artic Bug-Byte, Quicksilva Silversoft and JRS Macronics displayed its new disc drive for the ZX81

One or two American agents were seen at the show, trying to tie up deals to market and distribute software in the US for the Timex Sinclair 1000 This 2K version of the ZX81 went on sale through retail outlets in the US in September

Between micro-enthusiasts the generation gap is a thing of the past.

The main drawback with the *Personal Computer World* show is its scale Visiting all the stands would have been an impossible marathon Also, since the stands ranged from selling purely business applications to selling micro games some of the exhibitors found that only a small proportion of those visiting their stand were interested in their products

Jenny Kin the show's publicity manager, was however clearly delighted "It has had the largest audience of any microcomputer show in the world That is obviously a reflection of the strength of the industry in the UK

"In 1981 we had 16,000 visitors Now we have had over 45 000 The last five years has seen the show grow with the industry and it will continue to do so "As far as the exhibitors go, the cost of being at the *Personal Computer World* show is less than some others You get what you pay for You pay for the audience — and we spent a lot of money getting that audience in

The next *Personal Computer World* show juggernaut is already at an advanced stage of planning It will be held in September 1983 once again in the Barbican Centre

Microscene Brum

Out of the hot-house atmosphere of the Barbican and up to the relative calm of Birmingham's Bingley Hall and Microscene Brum 82

This show, on September 11, occupied only a small part of this vast draughty venue The remaining three-quarters of the hall was left to form a spacious rest area

Even though a large number of enthusiasts attended there was still plenty of room to get around and see what the 60 exhibitors had to offer

The event had a strong ZX and educational flavour since its organiser, Eric Deeson, also runs EZUG the Educational ZX Users Group Several Spectrums and one Dragon-32 were on display as was the Macronics ZX81 disc drive

On the whole, those exhibitors with ZX81 material did well and those with Spectrum wares did not One company in the latter group took less than £10 the whole day This was backuse, for some reason very few Spectrums have so far been delivered in the Birmingham area

Most people however exhibitors and visitors alike rated the fair a success Said Eric Deeson "We expected a thousand people to turn up We hoped for 2000 and we got 3500 We ran out of tickets at midday so, yes it went very well"



Alleys at the Barbican

An amalgam of Vic20 adventures



Novice adventurer: Mike Grace puts the bite on Dracula

Adventure games are claimed to be among the most popular games available for microcomputers. Having spent an exhausting weekend trying to escape the bite of the dreaded Dracula, exploring a Voodoo castle, racing against time to discover a timebomb ticking away in a nuclear plant and several other equally amazing feats of daring, I can quite see why.

For those people who don't know exactly *what* an adventure game is, or have heard the term but never seen a game in action, I will attempt a brief explanation. Adventure games are really computerised versions of role playing games, the best known of which is *Dungeons and Dragons* (abbreviated to *D&D* by those in the know). Unlike games such as Monopoly, there is no board as such. Instead, the players assume the roles of various characters in a fantasy story, for example a wizard, dwarf, witch or princess.

The players then have to carry out certain tasks, usually rescuing someone or finding some treasure. At the same time, the players have to cope with sundry nasty attempts by evil magicians, dragons or powerful spirits to remove them from the game. To try and help the characters in their quest, they can acquire magic spells and useful weapons along the way.

The real essence of the game is the preparation of a plot, and a series of labyrinthine tunnels or rooms, the exact location of which are known only by the 'dungeonmaster'. Thus, to take an example, the game could start with the dungeonmaster telling his group that they are in the grounds of a castle. There is a door in the wall ahead of them, and a huge ogre pursuing them with a club in his hand to smash them to pulp.

One of the group tries to open the door in the castle wall. The dungeonmaster replies that it is locked. Another of the group reveals that he has a magic key (acquired earlier in the game) which he tries in the lock. The dungeonmaster explains that the key fits, but will not turn. And so the game progresses.

Over the last 10 years or so *D&D* has built up a cult following. With the advent of the microcomputer it was obvious that someone would transfer the game into the sphere of machine-code language. Perhaps the best known of adventure game writers is Scott Adams. Five of his adventures have now been transferred onto cartridges by Commodore for the Vic20, hence my exhausting weekend.

I had heard about Scott Adams, but had never seen a game in action. So it was as a relatively green player that I sat down to the first game in the series — *Adventure Land*.

The packaging was more attractive than most games I have seen, (a sleeping dragon looking as if it should be adorning the cover of Tolkien's *The Hobbit*). I inserted the cartridge into the slot in the back of my expansion unit, and switched on. The first problem, that it did not work, was soon solved by removing the extra 16K of Ram that I usually keep permanently in my expansion unit — it might have been helpful if Commodore had added that precaution to their instructions — and I typed *Sys* 32592 to start.

In an adventure game the computer takes on the role of the dungeonmaster (or organiser). There are no graphics at all in these adventures, a feature I thought might spoil the game, but I can safely say that the fascination of trying to outwit the adventure more than compensates for the lack of pictures. In fact I was hooked almost instantly into the style of the game and quickly began to appreciate its versatility compared with the *Pacman/Space Invaders* type.

Having inserted the cartridge and keyed in the *Sys* command, the computer asks first if you want to restore a previous game. One of the features of these cartridges is that they allow you to save a game part way through and then load it back from the tape so that you can start where you left off. This is very valuable as the more skilled you become the longer it takes to try and solve the puzzle. (I should point out that I have not yet managed to solve any of the Commodore adventures completely.)

I typed in *No* and up came the welcome note plus the instructions (in green to help clarify your situation) that I was in a forest with trees and the words North, South, East, West This was followed by the instruction "What shall I do now? I gazed at the screen perplexed for a moment, and returned to the instruction manual

One of the notes of advice was to use instructions to the computer of two words only with words like *Climb Drop, Enter Examine* and *Take* to give the Vic a command I also discovered that if I typed *N* for *Go North* or *S* for *Go South*, using the key letters for the direction then the computer would move me to a new location

I returned to the screen and sat in slow thought for a few moments wondering which way to go — when inspiration struck me If I climbed a tree perhaps I would see which way was best So I typed *U* for up and was abruptly informed "Can I go that way! After a few abortive attempts to move in various directions I was suddenly inspired to write *Climb Tree* whereupon I was given a further instruction which told me but that would spoil your fun

As I moved through the game I began to get the hang of the main flaw in using a computer as a dungeonmaster, instead of as a real person it has a very limited dictionary so a lot of the time you are struggling to find the appropriate word I also found it very hard to keep my instructions down to two words, the first of which must be a verb

Whenever I was really stuck I would type *Help* — sometimes it *would* help and sometimes it would remain infuriatingly unhelpful

An example of this can be seen if I take the fifth adventure — called *The Count* and obviously inspired by Dracula I was in his castle and had found a dumb-waiter which I was sure would take me to a new part of the adventure, but when I typed *Climb In* (or something like that — I'm not telling you everything) all I was told was that I was inside a dumbwaiter I tried typing *Up* and *Down* to no avail I was told You can't go that way

Eventually, I typed *Help* and had to smile at the response which came back — "I know how to *Raise* and *Lower* this thing True enough, once I had keyed in *Raise Dumbwaiter* I was off into the adventure again

There are five adventures in this series The first, *Adventureland* is a true dungeons and dragons saga where the object is to discover 13 treasures and store them in a safe place The game starts in the forest, as I said, but soon moves underground — if you go wrong you can end up in hell itself The second game *Pirate Cove*, is another treasure hunt on a pirate island But this time you start inside a flat in London where there are several strange items in a room upstairs and a knowledge of carpentry can be of help(?)

In the third *Mission Impossible*, you are plunged into a race against time as a bomb threatens to explode in a nuclear power station This game comes complete with the tape-recorded instructions that the tv programme Mission Impossible used to start with including " this tape will self-destruct and is an ingenious and quite frustrating game The fourth *Voodoo Castle* I have not yet managed to even start to crack as yet (I know I can do it, but all attempts have failed miserably, and I've been destroyed whenever, so far I've attempted it) and involves trying to rescue a character called Count Christie from an evil curse And the fifth simply called *The Count*, is a chase story with Dracula trying to get you before you can get him

There is a delightful sense of humour running throughout the games, and the computer's response often made me laugh out loud I was particularly amused in the nuclear power plant when at a crucial moment a piece of paper fluttered to the floor When I picked it up and read it I was told to look for Adventure Number 4 at my favourite computer store

I found another advertisement in *Pirate Cove* and doubtless have others still to discover But the games can also be very frustrating, as in the case of being told an envelope contains a map and some keys but on writing *Open Envelope* I met with no success

My overall impression of the games is that they are extremely stimulating, far more addictive than any I have seen and well worth the money I have heard the criticism that once you have worked them out they have lost all value This may well be true but as far as I can see, it will be a very long time before I will have completely solved a game so I have many hours of pleasure ahead And I would imagine that after a year or so it should be possible to return to the game and start afresh

My children aged 8 and 12 have both become addicted It is pleasurable to sit in a family group around the screen and play together a factor missing from many computer games sadly so in my opinion To have the combined meeting of minds adds to the fun

My advice to the novice is to draw a map as described in the instructions right at the start — and to turn the game off and on again if stuck as the second time around it is surprising what new paths you can uncover More than that I will not say — it is up to you to find it out

I must return to *Castle Dracula*, for evening is drawing in and soon the Count will awake, and I'm still stuck in that dumbwaiter



Instruction booklets for the Vic20 adventure series

Open Forum

Open Forum is for you to publish your programs and ideas
It is important that your programs are bug free before you send them in. We cannot test all of them
Contributions should be sent to Popular Computing Weekly *Hobhouse Court,*
19 Whitcomb Street, London WC2H 7HF

How to contribute

Each week the editor goes through all the programs that you send to Open Forum in order to find the Program of the Week.

The author of that program will qualify for DOUBLE the usual fee we pay for published programs (The usual fee is £10.)

Presentation hints

Programs which are most likely to be considered for the Program of the Week will be computer printed and accompanied by a cassette.

The program will be well documented, the documentation being typed with a double spacing between each line

The documentation should start with a general description of the program and then give some detail of how the program has been constructed and of its special features

Listings taken from a ZX Printer should be cut into convenient lengths and carefully stuck down on to white paper, avoiding any creasing.

Please enclose a stamped, self-addressed envelope.

Superposition

on ZX81

Superposition is a program which runs comfortably in 4K of Ram on a ZX-81 The program demonstrates the physical principle of superposition which is concerned with wave motion and is a basic part of any physics 'A' level course and of some 'O' level courses

The program starts by defining the principle of superposition and proceeds with instructions on how to use the program to generate graphs of super-imposed waves and their corresponding data on the ZX printer (if available) The user is then invited to tell the computer how many waves he wishes to super impose and how many points he wishes to be plotted The relevant data for each wave is then entered one wave at a time

The data for each sine wave are the *amplitude* (the height of the wave) the *frequency* of the wave (the number of oscillations per second) and the *phase angle* (the value of the angle corresponding to the left-hand side of the screen)

When all the data has been entered the computer performs the kind of operation for which computers were originally designed, that is a massive amount of tedious number crunching in a relatively short period of time (even on a ZX81) The result is a graph representing the effect of super-imposing the effects of each wave

This is an ideal example of how computers can be used in education and can be used to demonstrate the concepts of interference beats (by taking waves of similar frequency) or simply wave motion in general

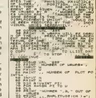
When the graph has been produced the user has the option of *Copying* the graph to the printer *Listing* the data for the waves on the printer or repeating the program with different data

The program has been written as concisely as possible in order to suit users who may only have 4K of *Ram* at their disposal This involves using all of the usual byte-saving techniques but only in areas of the program which can afford the extra time consumed by using *Code* and *Val* etc (i e not in the graph plotting routine)

Lines 5 to 36	are the explanation of how to use the program
Lines 40 to 70	input the number of waves and the number of plot points
Lines 75 to 135	are a *For Next* loop which take in the data for one wave every time a loop is made Note the use of *Sgn Pi*-1 in Line 75
Line 145	Puts the computer into *Fast* mode for speed if more than 100 points are to be plotted
Lines 150 to 190	are the routine which takes all of the wave data from array A and turns it into a graph
Lines 195 to 225	deal with what the user decides to do when the graph has been finished (*Copy* graph *List* data *Stop* or *Run* again)
Lines 230 to 235	*Copy* the graph and return for another input
Lines 240 to 280	*List* the data to the ZX printer
Lines 285 to 300	are the wait until the next key is pressed routine called during the introduction at the beginning of the program

The data for the waves is stored in a number of array A by lines 75 to 135 The data is stored in sections the first third of the array storing amplitudes the second third storing frequencies and the last third storing phase angles This means a multi-dimensional array is not needed

Line 175 checks that the y-coordinate of the plot point is in the range of the screen if not then the point is not attempted to be plotted one irritating feature of the ZX81 is that it will not ignore out-of-range *Plot* coordinates



```
   5 REM ****SUPERPOSITION****
  10 REM  (C)DAVID M. WEBB,1982
  15 SLOW
  20 PRINT "THIS PROGRAM ILLUSTRATES THE PHYSICAL PRINCIPLE OF SUPER-POSITION, WHICH STATES THAT THE NET DISPLACEMENT AT A GIVEN PLACE AND TIME CAUSED BY A NUMBER OF WAVES WHICH ARE TRAVERSING THE SAME SPACE IS THE SUM OF THE DISPLACEMENTS WHICH WOULD HAVE BEEN PRODUCED BY THE INDIVIDUAL WAVES SEPERATELY. ",,,"
  25 GOSUB VAL "285"
  30 PRINT "A GRAPH WILL BE DRAWN OF THE RESULTANT DISPLACEMENT AGAINST TIME, OVER AN INTERVAL OF ONE SECOND. THE SCREEN COVERS THE RANGE 21 TO -21 CENTIMETRES. WHEN THE GRAPH IS FINISHED,INPUT C TO COPY, D TO LLIST DATA, S TO STOP, N/L TO CONTINUE"
  35 GOSUB VAL "285"
  40 PRINT "NUMBER OF WAVES=",
  45 INPUT W
  50 PRINT W
  55 PRINT ,,"NUMBER OF PLOT POINTS=",
  60 INPUT N
  65 PRINT N
  70 DIM A(W*INT PI)
  75 FOR A=SGN PI TO W
  80 CLS
  85 PRINT "NUMBER ";A;" OUT OF ";W;" WAVES"
  90 PRINT ,,"AMPLITUDE(CM)=";
  95 INPUT A(A)
 100 PRINT A(A)
 105 PRINT ,,"FREQUENCY(HERTZ)=";
 110 INPUT A(A+W)
 115 PRINT A(A+W)
 120 PRINT ,,"PHASE ANGLE (RADS.)=";
 125 INPUT A(A+W*2)
 130 PRINT A(A+W*2)
 135 NEXT A
 140 CLS
 145 IF N>VAL "100" THEN FAST
 150 FOR X=SGN PI TO N-SGN PI
 155 LET P=X/(N-1)
 160 LET Y=0
 165 FOR A=1 TO W
 170 LET Y=Y+A(A)*SIN (2*PI*A(A+W)*P+A(A+2*W))
 175 IF ABS Y>21.75 THEN GOTO 190
 180 NEXT A
 185 PLOT P*63,21.75+Y
 190 NEXT X
 195 SLOW
 200 INPUT I$
 205 IF I$="C" THEN GOTO VAL "230"
 210 IF I$="D" THEN GOTO VAL "240"
 215 IF I$="S" THEN STOP
 220 CLS
 225 RUN CODE "9"
 230 COPY
 235 GOTO VAL "200"
 240 LPRINT ,,W;" WAVES ARE SUPERPOSED"
 245 LPRINT ,,N;" POINTS WERE PLOTTED"
 250 FOR A=SGN PI TO W
 255 LPRINT ,,"DATA FOR WAVE ";A
 260 LPRINT ,,"AMPLITUDE=",A(A);" CM"
 265 LPRINT ,,"FREQUENCY=",A(A+W);" HZ"
 270 LPRINT ,,"PHASE ANGLE=",A(A+2*W);" RADS"
 275 NEXT A
 280 GOTO VAL "200"
 285 IF INKEY$<>"" THEN GOTO VAL "285"
```

PROGRAM OF THE WEEK

```
280 IF INKEY$= " THEN GOTO VAL
290
295 CLS
300 RETURN
```

```
THIS PROGRAM ILLUSTRATES THE
PHYSICAL PRINCIPLE OF SUPER-
POSITION, WHICH STATES THAT
 THE NET DISPLACEMENT AT A GIVEN
PLACE AND TIME CAUSED BY A
NUMBER OF WAVES WHICH ARE
TRAVERSING THE SAME SPACE IS THE
SUM OF THE DISPLACEMENTS WHICH
WOULD HAVE BEEN PRODUCED BY THE
INDIVIDUAL WAVES SEPERATELY.

A GRAPH WILL BE DRAWN OF THE
RESULTANT DISPLACEMENT AGAINST
TIME, OVER AN INTERVAL OF ONE
SECOND. THE SCREEN COVERS THE
RANGE 21 TO -21 CENTIMETRES.
WHEN THE GRAPH IS FINISHED INPUT
 C  TO COPY
 D" TO LLIST DATA
"S  TO STOP ,
N/L TO CONTINUE

NUMBER OF WAVES=2
NUMBER OF  PLOT POINTS=240
```

```
NUMBER 1 OUT OF 2 WAVES
AMPLITUDE(CM.)=10
FREQUENCY(HERTZ)=1
PHASE ANGLE (RADS.)=0

NUMBER 2 OUT OF 2 WAVES
AMPLITUDE(CM.)=10
FREQUENCY(HERTZ)=5
PHASE ANGLE (RADS.)=3.1415927
```

```
2 WAVES ARE SUPERPOSED
200 POINTS WERE PLOTTED

DATA FOR WAVE 1
AMPLITUDE=10 CM.
FREQUENCY=1 HZ.
PHASE ANGLE=0 RADS.

DATA FOR WAVE 2
AMPLITUDE=10 CM.
FREQUENCY=5 HZ.
PHASE ANGLE=3.1415927 RADS.
```

```
2 WAVES ARE SUPERPOSED
2000 PRINTS WERE PLOTTED

DATA FOR WAVE 1
AMPLITUDE=10 CM.
FREQUENCY=17 HZ.
PHASE ANGLE=0 RADS.

DATA FOR WAVE 2
AMPLITUDE=10 CM.
FREQUENCY=20 HZ.
PHASE ANGLE=3.1415927 RADS.
```

Superposition
by David Webb

League Table

on BBC Micro

For all the football enthusiasts this listing is for the Football League Division One. It can be adapted for any football league. All that need changing are the team names in the Data statements in lines 730-760.

These should be placed in alphabetical order and based on this order the teams are numbered from 0 on (league total). In line 30 J% is set the number of teams in the league less one.

The names given to the files in lines 130 and 600 can be changed to suit. The title can be changed in line 430. The variable names are mainly self-explanatory. The original name *Draw* had to be substituted by D because of the confusion with the command word *Draw*.

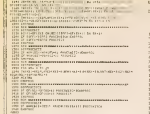
When entering the team numbers and scores in lines 270-290 a check is kept that duplicate team numbers are not entered and the program allocated points and goals scored to the correct teams. These are sorted before printing out the new league tables. Should a printer not be available then lines 420, 440 and 520 will not be required.

The team statistics are sorted back into alphabetical order ready for reading back into the program on the next run.

```
   5 MODE 0
  10 INPUT "IS THIS THE FIRST RUN OF THE PROGRAM ?   TYPE Y FOR
YES AND N FOR"N"'I$:IFI$="N" OR I$="Y"GOTO20 ELSE GOTO10
  20 J%=21
  30 INPUT"HOW MANY RESULTS TO RECORD?"C%
  40 DIM TN(J%),PLAYED(J%),WON(J%),D(J%),LOST(J%),GF(J%),GA(J%),
PTS(J%),N(J%),NAME$(J%)
  60 IF I$="Y"PROCINIT:GOTO250
 110 *FX 137,1
 120 *FX 7,3
 130 X=OPENIN ("LEAGUE1")
 140 FOR N%=0 TO J%
 150 INPUT#X,TN(N%),PLAYED(N%),WON(N%),D(N%)
,LOST(N%),GF(N%),GA(N%),PTS(N%)
:NEXT N%
 240 CLOSE#X
 245 *FX 137,0
 250 FOR N%=0 TO J%:READ NAME$(N%):N(N%)=0:NEXT N%
 260 FOR H%=1 TO C%
 270 INPUT "HOME TEAM NUMBER AND SCORE "TM1,SC1
 280 IF N(TM1)=1 THEN 270 ELSE N(TM1)=1
 290 INPUT "AWAY TEAM NUMBER AND SCORE "TM2,SC2
 300 IF N(TM2)=1 THEN 290 ELSE N(TM2)=1
 310 PLAYED(TM1)=PLAYED(TM1)+1:PLAYED(TM2)=PLAYED(TM2)+1:GF(TM1)
=GF(TM1)+SC1:GA(TM1)=GA(TM1)+SC2:GF(TM2)=GF(TM2)+SC2:GA(TM2)=
GA(TM2)+SC1
 315 IF SC1>SC2 PROCHWIN:GOTO350
 320 IF SC1=SC2 PROCDRAW:GOTO350
 330 PROCLOSE
 340 NEXT H%
 350 S%=0:X%=0
 360 REPEAT
 370 IF PTS(S%)<PTS(S%+1) PROCSWITCH:GOTO390
 380 IF PTS(S%)=PTS(S%+1) PROCGOAL
 390 S%=S%+1
 400 UNTIL S%=J%
 410 IF X%=1 GOTO350
 420 VDU2,1,27,1,14
 430 PRINT "LEAGUE DIVISION ONE"
 440 VDU1,27,1,15
 450 PRINT
 460 PRINT"   TEAM NAME           P   W   D   L   F   A  PTS"
                                                GOALS
 470 FOR N%=0 TO J%
 480 M%=TN(N%)
 490 PRINT(N%+1);TAB(4);NAME$(M%);TAB(20);PLAYED(M%);TAB(24);
WON(M%);TAB(28);D(M%);TAB(32);LOST(M%);TAB(37);GF(M%);TAB(40);
GA(M%);TAB(44);PTS(M%)
 510 NEXT N%
 520 VDU3
 530 S%=0:X%=0
 540 REPEAT
 550 IF TN(S%)>TN(S%+1) PROCSWITCH
 560 S%=S%+1
 570 UNTIL S%=J%
 580 IF X%=1 GOTO530
 585 *FX137,1
 590 *FX 7,3
 600 X=OPENOUT ("LEAGUE1")
 610 FOR N%=0 TO J%
 620 PRINT#X,TN(N%),PLAYED(N%),WON(N%),D(N%),
LOST(N%),GF(N%),GA(N%),PTS(N%):NEXT N%
 710 CLOSE#X
 715 *FX 137,0
 720 END
 730 DATA ARSENAL,"ASTON VILLA","BIRMINGHAM",BRIGHTON,
COVENTRY,EVERTON
 740 DATA IPSWICH,LIVERPOOL,LUTON,"MAN CITY","MAN UTD",
NORWICH
 750 DATA "NOTTM FOREST","NOTTS COUNTY",SOUTHAMPTON,"STOKE
CITY",SUNDERLAND
 760 DATA SWANSEA,TOTTENHAM,WATFORD,"WEST BROM","WEST HAM"
 770 REM *************************************************
 780 DEFPROCHWIN
 790 WON(TM1)=WON(TM1)+1:PTS(TM1)=PTS(TM1)+3
 800 LOST(TM2)=LOST(TM2)+1
 810 ENDPROC
 890 REM *************************************************
 900 DEFPROCDRAW
 910 D(TM1)=D(TM1)+1:PTS(TM1)=PTS(TM1)+1
 920 D(TM2)=D(TM2)+1:PTS(TM2)=PTS(TM2)+1
1010 ENDPROC
1020 REM *************************************************
1030 DEFPROCLOSE
1040 LOST(TM1)=LOST(TM1)+1
1050 WON(TM2)=WON(TM2)+1:PTS(TM2)=PTS(TM2)+3
1100 ENDPROC
1110 REM *************************************************
1120 DEFPROCSWITCH
```

to next page

from previous page





LEAGUE DIVISION ONE

DATE - 28 AUG 82

	TEAM NAME	P	W	D	L	GOALS F	GOALS A	PTS
1	MAN.UTD	1	1	0	0	3	0	3
2	SUNDERLAND	1	1	0	0	3	1	3
3	LIVERPOOL	1	1	0	0	2	0	3
4	WATFORD	1	1	0	0	2	0	3
5	MAN.CITY	1	1	0	0	2	1	3
6	NOTTM FOREST	1	1	0	0	2	1	3
7	STOKE CITY	1	1	0	0	2	1	3
8	COVENTRY	1	1	0	0	1	0	3
9	LUTON	1	0	1	0	2	2	1
10	TOTTENHAM	1	0	1	0	2	2	1
11	BRIGHTON	1	0	1	0	1	1	1
12	IPSWICH	1	0	1	0	1	1	1
13	NOTTS COUNTY	1	0	1	0	0	0	1
14	SWANSEA	1	0	1	0	0	0	1
15	ARSENAL	1	0	0	1	1	2	0
16	NORWICH	1	0	0	1	1	2	0
17	WEST HAM	1	0	0	1	1	2	0
18	SOUTHAMPTON	1	0	0	1	0	1	0
19	ASTON VILLA	1	0	0	1	1	3	0
20	EVERTON	1	0	0	1	0	2	0
21	WEST BROM	1	0	0	1	0	2	0
22	BIRMINGHAM	1	0	0	1	0	3	0

League Table
by B Gagg

Mystery

on Vic 20

Most Vic20 users probably like myself, assess a program by looking at the listing and getting a rough idea of what it is going to do But this program I have called Mystery Program as I am confident that no Vic20 users could look at the listing and have any idea what the program does

The program as it stands will only be run on an unexpanded Vic due to the start of basic location (4096) Adjusting line 40 to For P=1024 to 1214 should make it run with a 3K expansion

The program is only short so don't expect miracles but the way in which it works should be of interest to some and is a basis for future experiment Lines 10 and 20 can contain any letters but there must be at least two complete lines of them (160)

Save the program before you run it because if you have made a mistake you'll have to power down and start again Now key it in and see what it does

```
10 REM XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXX
20 REM XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXX
40 FORP=4096TO4206 READD POKEP,D NEXT
50 DAT98,61,16,1,8,151,51 54,56,52-57,44,58,58 153,34,147,5,17,17,17,17,17,17,17
,17,17
60 DATA29,85,83,82,32,67,85,82,82,79,82,32,67,78,78,84,82,79,76,82,38,58,128,85,
178,49
70 DATA164,58,48,48 48,58,138,8,56 16,2,0,55,178,55,57,48 48 58,86 178,49,49,58,
66 178,58
90 DATA49,54,52,56,129,74,178,49,164,58,32,56,133,58,138,8,146,16,3,0,151,66,178
,181,40
98 DATA58,58,172,167,48,49,41,41,44,52,58,58,151,55,178,66 44,51,58,58,153,56,13
9,134 48
100 DATA49,57,55,41,178,51,49,179 56,177,49,167,56,178,56,171,49,8,174,16,4 8,13
9 134
110 DATA48,49,57,55,41,178,58 51,175,66,179,58,49,167,66,178,58,178,49,8,189 16,
5 0,151
120 DATA65,178,66,44,58 58,58,137,51,8,0,8

READY.
```

Mystery
by Andy Horrell

Letter Writer

on Vic 20

This program offers a word processing capability for the home user Whilst not offering a full processing system as in professional systems, it does provide an easy and workable system

Upper and lower case is used with up to 30 lines of text available not including address heading and usual letter ending

Lines 6 to 13 and 3265 to 3269 are an optional motif which the user can design to his own liking As lines of text are entered a score shows how many have been entered so far The user should put his address in the appropriate places shown in the listing and they will appear automatically when run

The user enters his lines when asked, up to 40 characters of text per line Should he enter too many characters the cursor will return to the beginning of that line, the user should then move the cursor along and shorten that line to less than 40 characters

The last line typed is then displayed with an option to correct if required Corrections cannot be made to that particular line after this chance When all message text has been entered, but before yours etc' — the user enters 'xx as one would enter a normal sentence He can then enter yours faithfully and hit return.

At this point the whole letter (other than personal motif) is displayed on the screen If you have typed in a large letter use 'control' to view it as it scrolls up the screen When viewed, the user hits any key and the printout option is displayed If printer accepted by user the whole letter is then printed on the Vic printer (GP80) in double size text (hence 40 characters per line) After printing an option for second copy is offered

from previous page

```
7 POKE36869,242
8 PRINT "
9 D1$=" 
10 D2$=" 
11 D3$=" R.BARTON."
12 D4$=" 
13 D5$=" 
20 PRINT"ADDRESS IS AUTOMATIC"
25 PRINT"
30 S$="YOUR STREET"
40 T$="TOWN"
50 CO$="COUNTY"
70 PRINT"DATE?"
80 INPUTDA$
90 PRINT"1ST.SENTENCE?"
99 DIMM$(30)
100 X=1
105 PRINT"SENTENCE NO. "X
120 PRINT" ":INPUTM$(X)
130 IFM$(X)="XX"THEN472
132 IFLEN(M$(X))>38THEN120
133 PRINT"IS THIS TO BE ACCPTD?"
134 PRINT" ":PRINTM$(X):INPUTM$(X)
135 X=X+1
140 PRINT" ":GOTO105
472 PRINT"ENDING...."
473 INPUTE$
900 PRINT"SIGNED...."
920 INPUTSI$
1000 GOTO2999
2999 PRINT"
3000 PRINTTAB(21-LEN(S$))S$
3002 PRINTTAB(21-LEN(T$))T$
3004 PRINTTAB(21-LEN(CO$))CO$
3006 PRINTTAB(21-LEN(DA$))DA$
3020 FORX=1TO30
3021 IFM$(X)="XX"THEN3099
3022 PRINTM$(X)
3025 NEXTX
3099 PRINTE$
3100 PRINT"  "TAB(21-LEN(SI$))SI$
3200 GETA$:IFA$=""THEN3200
3210 PRINT"DO YOU REQUIRE
3220 PRINT"PRINTOUT/
3230 PRINT"Y/N"
3240 GETA$:IFA$=""THEN3240
3250 IFA$="N"THENEND
3260 OPEN1,4
3265 PRINT#1,CHR$(14)CHR$(16)"10"D1$
3266 PRINT#1,CHR$(10)CHR$(16)"10"D2$
3267 PRINT#1,CHR$(10)CHR$(16)"10"D3$
3268 PRINT#1,CHR$(10)CHR$(16)"10"D4$
3269 PRINT#1,CHR$(10)CHR$(16)"10"D5$
3280 PRINT#1,CHR$(10)
3281 PRINT#1,CHR$(10)
3287 PRINT#1,CHR$(146)CHR$(17)CHR$(16)"4
0"S$
3290 PRINT#1,CHR$(17)CHR$(16)"40"T$
3300 PRINT#1,CHR$(17)CHR$(16)"40"CO$
3310 PRINT#1,CHR$(17)CHR$(16)"40"DA$
3312 PRINT#1,CHR$(10)
3313 PRINT#1,CHR$(10)
3315 FORX=1TO30
3317 IFM$(X)="XX"THEN3417
3320 PRINT#1,CHR$(17)M$(X)
3325 NEXTX
3417 PRINT#1,CHR$(10)
3418 PRINT#1,CHR$(10)
3419 PRINT#1,CHR$(17)CHR$(16)"30"E$
3420 PRINT#1,CHR$(10)
3425 PRINT#1,CHR$(17)CHR$(16)"45"SI$
3430 CLOSE1,4
3440 PRINT"ANOTHER COPY?
3450 GETA$:IFA$=""THEN3450
3460 IFA$="Y"THEN3260
3470 CLOSE1:END
```

Letter Writer
by Richard Barton

Alien

on Spectrum

This is a fast-moving arcade style game Instructions are in the program The aliens move up the screen using the automatic scroll and the 'scroll counter' is *Poked* (*Poke* 23692 255) for smooth graphics

The difficulty may be increased by changing line 65 This line chooses a random square beneath your ship — if it finds a # (centre of alien) and then a random number greater than 65 it fires To make it more difficult change the 65 to 2 Or to make it easier change it to 8

In line 100 the ship is made of a "(Graphic A)(INV#)(Graphic B)" My high score is 32

```
1 LET h=0: LET n$=""
10 LET n=0: FOR f=0 TO 7: LET n=n+2↑f: POKE USR "a"+f,n: NEXT f
20 LET n=0: FOR f=7 TO 0 STEP -1: LET n=n+2↑f: POKE USR "b"+7-f,n: NEXT f
30 GO SUB 660: LET a=10: LET b=0: LET s=0
40 BORDER 0: PAPER 0: INK 7: CLS
45 POKE 23692,255
50 LET a=a+(INKEY$="8" AND a<30)-(INKEY$="5" AND a>0)
60 PRINT INK 5;AT 0,a;"  "
65 LET s=RND*28: IF SCREEN$ (s,a+1)="#" THEN IF RND>.65 THEN GO TO 400
70 IF s>0 THEN GO TO 200
75 IF INKEY$="" THEN GO TO 100
80 IF RND>.9 THEN GO TO 50
90 PRINT AT 21,0
95 POKE 23692,-1
100 PRINT TAB RND*27; INK 1;"  ": IF SCREEN$ (0,a+1)<>" " THEN GO TO 300
110 GO TO 50
190 LET b=1: LET b1=a+1: LET a=0
200 PRINT AT b,b1;" ": LET b=b+1: LET a1=a1+a
210 IF b1=-1 THEN LET b1=31
215 IF b1=32 THEN LET b1=0
220 IF b>21 THEN LET b=0: GO TO 50
230 LET a$=SCREEN$ (b,b1)
240 RESTORE
250 IF a$="#" THEN GO TO 350
255 IF a$<>" " THEN GO TO 250
260 PRINT AT b,b1;"": GO TO 50
265 LET a=(SCREEN$ (b,b1+1)="#")-1: IF a=0 THEN LET a=1
285 LET b=b+1: LET a1=b1+a: IF b1>21 THEN LET b=0: GO TO 50
290 GO TO 220
350 RESTORE: FOR f=1 TO 10: READ q$: FOR g=1 TO 3: PRINT AT b,a+1,q$(g): NEXT g: NEXT f
310 CLS: PRINT "You scored ";s
311 IF s>h THEN PRINT "You have the top score of ";s: INPUT "Your name",n$: LET h=s
320 IF INKEY$="" THEN GO TO 320
330 GO TO 30
350 FOR f=1 TO 10: READ q$: PRINT AT b,b1-1;q$: NEXT f: LET s=s+1: LET b=0: GO TO 50
400 FOR f=a TO 0 STEP -1: PRINT AT f,a+1;"+": BEEP .5,f/10: NEXT f: GO TO 300
500 CLS: PRINT "____________ALIEN STORM"
510 PRINT "Your ship has drifted into a fleet of oncoming aliens you must"
520 PRINT "(1) Avoid their path"
530 PRINT "(2) Blast them with your gun"
540 PRINT "BEWARE-they shoot only if you stay above them for too long (they never miss!)"
550 PRINT FLASH 1;"PRESS ANY KEY"
560 PAUSE 0: CLS
570 PRINT "CONTROLS..."
580 PRINT "5=left 8=right, space=fire"
590 PRINT "HINT-aim for the centre of the alien for your beams bounce off their bodies!"
600 PRINT FLASH 1;"PRESS ANY KEY"
610 PAUSE 0: CLS
620 PRINT "The high score = ";h
640 PRINT "by ";n$
650 PRINT FLASH 1;"PRESS ANY KEY"
660 PAUSE 0: CLS: RETURN
990 DATA "***","+++","-"-"","$$$",",",";","???","*+-","!!"
```

Alien
by Nick Wilson

```
1 REM CHARACTER CHANGE FOR 16K ZX SPECTRUM
2 REM © G.R.Tapp 27/8/1982
5 CLEAR 31574
10 FOR a=31575 TO 31820: POKE a,255: NEXT a
20 FOR a=31621 TO 31767
30 LET x=PEEK (15616+(a-31621))
40 POKE a,x: NEXT a
50 FOR a=31768 TO 31847: READ x: POKE a,x: NEXT a
55 REM Numbers
60 DATA 0,126,78,74,82,98,126,0,0,0,0,0,0,0,126,2,2,126,64,126,0,0,126,2,2,126,2,126,0,0,72,72,72,126,8,8
65 DATA 0,126,64,64,126,2,126,0,0,126,64,64,126,66,126,0,0,126,2,2,2,2,0,0,126,66,66,126,66,66,126,0,0,126,66,66,126,2,2,0
70 FOR a=32849 TO 32895
80 LET x=PEEK (15616+(a-31621))
90 POKE a,x: NEXT a
100 FOR a=32896 TO 32982
110 READ x: POKE a,x: NEXT a
120 DATA 0,126,66,126,66,66,66,0,0,124,66,124,66,66,124,0,0,126,64,64,64,126,0
130 DATA 0,124,66,66,66,66,124,0,0,126,64,126,64,64,126,0,0,126,64,126,64,64,64,0
140 DATA 0,126,66,64,78,66,126,0,0,66,66,126,66,66,66,0,0,0,0,0,0,0,0,0,0,0,0,0,0,126,0,0,0,0,66,0
160 DATA 0,64,64,64,64,126,0,0,66,66,0,64,64,64,64,64,126,0,0,126,66,66,66,66,0,0,126,66,66,66,66,0
165 DATA 0,126,66,66,66,66,126,0,0,126,66,126,64,64,64,0,0,126,66,66,74,78,126,0
170 DATA 0,126,66,126,72,66,66,0,0,126,64,126,2,2,126,0,0,126,0,0,0,0,0,0,66,66,66,66,66,126,0
180 DATA 0,66,66,36,36,24,24,0,0,98,90,90,90,90,126,0,0,66,36,24,36,66,0,0,66,66,126,2,2,126,0,0,126,2,4,8,16,126,0
200 FOR a=32984 TO 33080
210 LET x=PEEK (15616+(a-31621))
220 POKE a,x: NEXT a
```

Character Change
by Gavin Tapp

Character Change

on Spectrum

ZX Character change converts all the standard ZX numbers and capital letters to more digital styled ones The program works by poking part of the *Rom* into the top of the *Ram* but changing some of the characters as it does so

After entering the listing type *Run* If nothing has gone wrong so far type in *Poke* 23606 88 *Poke* 23607 123 then list If all has gone well you will see that all the numbers and capitals will look digital

You can now *New* the program (don't worry, only the program will disappear not the characters) To save the characters use *Save* 'Name of program' *Code* 31574 1025 To load in use *Clear* 31574 *Load* '' *Code*

You can now do what you like with the new numbers and letters and change between the two by using *Poke* 23606 88 *Poke* 23607 123 to go into new character mode or use *Poke* 23606 0 *Poke* 23607 60 to return back to normal letters and numbers

Minipro

on BBC

This program sets up the user definable keys of a BBC microcomputer so that it can be used as a simple word processor. Text is entered into a program so that the existing editing facilities of the computer can be used to make corrections. Text can be printed by *Running* it to the printer and *Saved* on cassette.

Far from idiot proof it is intended for the computer hobbyist as a useful aid to letter writing, or as here, to writing program documentation.

Program notes

Lines 60 to 150 set up the red keys —

KEY0 removes the program, starts the auto line numbering, and inserts the first print statement so that typing can commence after a single keystroke.

KEY1 enables the move to a new line as easily as if you were pressing the carriage return on a typewriter.

KEY2 is used instead of KEY1 if the next line requires a *Tab*.

KEY3 enables the *Tab* arising from KEY2 to be closed with a single key.

KEY4 is used when you want to start editing and after *Escaping* from the text program. After pressing this key you edit as you would any program on the BBC micro.

The listing is in page mode, and you therefore have to use *Escape* before editing, and *Shift* to scroll. If you want to restart the text program use *Auto* and the appropriate line number followed by *Print*.

When finished it is best to make sure the printer is off (KEY6), and *Run* the text program to see on the screen precisely how it will eventually appear on paper. KEY4 will then re-enter edit mode if required.

KEY5 is used to print the text, so make sure the editing is complete and the printer is correctly set up before you press it.

KEY6 switches the printer off. Make sure you press this after printing is finished otherwise you may get some unwanted additions to your text when you next use the keyboard.

KEY7 is pressed after *Escaping* from the text program to see how much memory is left.

KEY8 gives 10 spaces for convenient starts to new paragraphs etc.

KEY9 gives 58 spaces for convenient addressing at the top R H S of letters.

Mode 3 is used to get a relatively easy visual indication of the line length for the 80 column printer being used.

Lines 170 to 200 print a reminder of the key definitions and set up a full width text window for the display.

Line 210 speeds up the cursor for editing purposes.

Main difficulty

The main difficulty is in deciding when to use a new line. You can get three full printed lines on each program line before the *Beep* tells you the line is full. Then you can delete the last word and use KEY1 or KEY2 to start the new line. Easy enough, but the full program line cramps your style on editing. It is probably best to have 2 lines of print for each program line.

Any ideas for improvements would be welcome. It would be interesting to see how many extra features can be added without using any *Ram*. There could be scope for machine code routines down in the operating system area.

To operate Minipro simply *Load* and *Run*. If you are starting a new text commence with KEY0, otherwise *Load* the text program you intend to modify and commence with KEY4. If you are not using a monitor and find the *Mode3* text difficult to read accurately, you can always do a final check-run in *Mode7* before printing.

```
L.
   10REM ***MINIPRO***
   20REM MINIATURE WORD PROCESSOR
   30REM BY C.R.WOODINGS
   40REM VERSION 1.1 / 30 AUGUST 1982
   50REM NEEDS A MODEL B BBC MICROCOMPUTER
   60*KEY0 NEW!M!CAUTO!MP."
   70*KEY1 """!MP."" "
   80*KEY2 ""'!MP.TAB("
   90*KEY3 )"
  100*KEY4 !C!NLIST!M
  110*KEY5 !ORUN!B!M
  120*KEY6 !C
  130*KEY7 V.11:DIMP%-1:P.(HIMEM-P%);" BYTES LEFT
"!M
  140*KEY8 "          "
  150*KEY9 "                                                          "
  160MODE3:COLOUR0:COLOUR129:CLS
  170PRINT"                                    ***MINIPRO***"'
  180PRINT"  f0=Start; f1=Newline; f2=Newline with Tab;
f3=CloseTab; f4=Edit; f5=Print"
  190PRINT"       f6=Printer Off;  f7=Available Memory;
f8=TAB(10);  f9=TAB(55)"
  195PRINT' **********************************************
*************************************"
  200VDU 28,0,24,79,5
  210*FX 12,3
  220END
```

Minipro
by Chris Woodings

Tennis

on Spectrum

This is a two-player game which will run on a 16k Spectrum. The idea of the game is to destroy as many bricks as possible in your opponent's wall with the ball during each rally whilst protecting your own wall.

Each rally lasts for twenty strokes and there are six rallies to one game. Returning the ball safely scores five points and destroying one brick scores ten.

If the ball returns from the wall through your bat then part of your bat will disappear. It will return when the bat is moved. Instructions showing how to move the bats are given in the program.

In order to detect the bat and the bricks the program uses the *Attr* function and so if any changes are made to the colours used in the program then it may be necessary to change the corresponding *Attr* functions.

Program notes

Lines	
10-40	Set up the user defined graphics for the bricks and the ball
50-150	Set up the playing area with each brick having a random colour
210	Reverses the direction of entry of the ball for successive rallies
240-250	Put the ball into play in a random direction from a random position near the centre of the screen
900-960	Read the keyboard using the *In* function in order to move the bats. This command, unlike *Inkeys*, will still work if more than one key is depressed on the keyboard. Thus, both players can move their bats simultaneously
1000-1090	Decide at which of the three possible angles the ball will rebound from the bat. This depends on where the ball hits the bat. If the ball misses the bat it will carry on to the wall
1200-1300	Determine whether the ball has hit a brick in the wall and if so will remove it and increment the score
2000-2350	Move the bats up and down according to the input as read from the keyboard

Graphics Notes

Lines	
10	Graphics A
20	Graphics B
30	Graphics C
60	Graphics shifted B
150	Graphics 5 and graphics shifted 5

```
   1 REM
   2 REM
   3 REM
   4 REM
   8 BORDER 7
  10 FOR x=0 TO 7: READ a: POKE
USR "A"+x,a: NEXT x
  20 FOR x=0 TO 7: READ a: POKE
USR "B"+x,a: NEXT x
  30 FOR x=0 TO 7: READ a: POKE
USR "C"+x,a: NEXT x
  40 DATA 0,126,126,126,126,126,
126,126,126,126,126,126,126,126,126,
126,0,0,24,60,126,126,60,24,0
  44 GO TO 4000
  45 LET y=1: LET lscore=0: LET
rscore=0
  50 FOR z=1 TO 6
  60 PRINT AT 0,0;lscore;AT 0,12
RALLY ";z;AT 0,27;rscore
  90 PRINT AT 1,0; INK 5;"
 100 FOR a=2 TO 20 STEP 2
 105 INK (RND*3)+4: PAPER RND*4
 110 PRINT AT a,31;"A";AT a+1,31
;"
 115 INK (RND*4)+4: PAPER RND*4
 120 PRINT AT a,0;"A";AT a+1,0;"
"
 130 NEXT a
 135 PAPER 7: INK 0
 140 LET bleft=11: LET brightbat
=11
 150 FOR a=10 TO 12: PRINT AT a,
2; INK 1;"█";AT a,29; INK 1;"█"
: NEXT a
 210 LET y=-y
 220 LET intx=y
 230 LET inty=INT (RND*3)-1
 235 LET rally=0
 240 LET ballx=INT (RND*10)+11:
LET bally=INT (RND*10)+6: BEEP
.2,10
 250 PRINT AT bally,ballx;"C"
 255 GO SUB 900
 260 IF ballx=3 AND intx=-1 THEN
 GO SUB 1000
```

```
261 IF ballx<25 AND intx=1 THEN
 GO SUB 1000
265 IF ballx=1 THEN GO SUB 1200
268 IF ballx=30 THEN GO SUB 130
0
270 LET ballx=ballx+intx: LET b
ally=bally+inty
275 IF bally<3 OR bally>21 THEN
 GO SUB 1100
285 PRINT AT bally-inty,ballx-i
ntx
290 IF rally>=10 THEN GO TO 300
0
300 GO TO 200
899 STOP
900 IF IN 64510=254 THEN GO SUB
2000
910 IF IN 65022=254 THEN GO SUB
2100
920 IF IN 57342=254 THEN GO SUB
2200
930 IF IN 49150=253 THEN GO SUB
2300
940 RETURN
1000 IF bally=bleft THEN LET int
y=0
1010 IF bally=bleft+1 THEN LET i
nty=1
1020 IF bally=bleft-1 THEN LET i
nty=-1
1025 IF ATTR (bally,ballx-1)=56
THEN GO TO 1030
1030 LET intx=-intx: BEEP .03,30
: LET lscore=lscore+5: PRINT AT
0,2;lscore
1035 LET rally=rally+1
1040 RETURN
1050 IF bally=bright THEN LET in
ty=0
1060 IF bally=bright+1 THEN LET
inty=1
1070 IF bally=bright-1 THEN LET
inty=-1
1075 IF ATTR (bally,ballx+1)=56
THEN GO TO 1080
1080 LET intx=-intx: BEEP .03,30
: LET rscore=rscore+5: PRINT AT
0,27;rscore
1085 LET rally=rally+1
1090 RETURN
1099 STOP
1100 LET inty=-inty
1110 LET bally=bally+2*inty
1120 RETURN
1200 IF ATTR (bally,0)=56 THEN G
O TO 1240
1210 LET hit=bally
1220 IF hit/2-INT (hit/2)<.1 THE
N LET hit=hit+1
1230 PRINT AT hit,0;" ";AT hit-1
,0;
1235 BEEP .1,-20: LET rscore=rsc
ore+10: PRINT AT 0,27;rscore
1240 LET intx=-intx
1290 RETURN
1300 IF ATTR (bally,31)=56 THEN
GO TO 1340
1310 LET hit=bally
1320 IF hit/2-INT (hit/2)<.1 THE
N LET hit=hit+1
1330 PRINT AT hit,31;" ";AT hit-
1,31;
1335 BEEP .1,-20: LET lscore=lsc
ore+10: PRINT AT 0,2;lscore
1340 LET intx=-intx
1350 RETURN
2000 IF bleft=3 THEN GO TO 2050
2010 LET bleft=bleft-1
2020 FOR a=bleft-1 TO bleft+1: P
RINT AT a,0; INK 1;"█": NEXT a
PRINT AT bleft+2,0;" "
2050 RETURN
2100 IF bleft=20 THEN GO TO 2150
2110 LET bleft=bleft+1
2120 FOR a=bleft-1 TO bleft+1: P
RINT AT a,0; INK 1;"█": NEXT a:
PRINT AT bleft-2,0;" "
2150 RETURN
2200 IF bright=3 THEN GO TO 2250
2210 LET bright=bright-1
2220 FOR a=bright-1 TO bright+1:
 PRINT AT a,29; INK 1;"█": NEXT
a: PRINT AT bright+2,29;" "
2250 RETURN
2300 IF bright=20 THEN GO TO 235
0
2310 LET bright=bright+1
2320 FOR a=bright-1 TO bright+1:
 PRINT AT a,29; INK 1;"█": NEXT
a: PRINT AT bright-2,29;" "
2350 RETURN
3000 FOR f=1 TO 10: FOR g=16 TO
14: BEEP .01,g: NEXT g: NEXT f
3010 PRINT AT 5,0;"END OF RALLY
 ";x;AT 8,7;"PRESS ANY KEY TO";AT
 11,11;"CONTINUE"
3020 IF INKEY$="" THEN GO TO 302
0
3025 CLS
3030 NEXT x
3040 PRINT AT 10,0;"DO YOU WANT
TO PLAY AGAIN? (y/n)"
3045 IF INKEY$="" THEN GO TO 304
5
3050 IF INKEY$="y" THEN GO TO 50
50
3060 STOP
3999 CLS
4000 PRINT AT 0,0;"The idea of t
he game is to destroy your op
ponent's wall with the ball wh
ilst protecting your own by de
flecting the ball with your ba
t. Each game consists of six
rallies each of twenty strokes
4010 PRINT AT 14,0;"Q -bat up
 P -bat up";AT 16,0;"A -b
at down L -bat down"
4020 PRINT AT 20,0;"press any ke
y"
4030 IF INKEY$="" THEN GO TO 403
0
4040 CLS : GO TO 45
```

Tennis
by Colin Leach

Bad Feature

on Dragon

One bad feature of the Dragon is that if a key is held down for any length of time, the function *Inkey$* will start to return a null string after a short interval. This can be a nuisance when testing for key presses in games programs etc, but can be overcome by using *Peek* (337) instead.

This will return a value for each key as long as that key is held down, or 255 if no key is being pressed. To find out which values are produced by which keys, enter the program line

10 ? peek (337): GOTO 10

and make a note of the value produced for each key by pressing different keys down. E.g. "A"=251, "7"=254, "8"=253.

Some of the values are duplicated for separate keys, so it may be necessary to use the 8 bytes after 337, e.g. to test for the cursor control keys:

- ↑ gives PEEK (341)=223
- ↓ gives PEEK (342)=223
- ← gives PEEK (343)=223
- → gives PEEK (344)=223

As an example of use in a program, instead of

10 IF INKEY$="A" THEN GOTO 100

use

10 IF PEEK (337)=251 THEN GOTO 100

Also, *Peek* (135) will return the Ascii value of the last key pressed, even if that key has since been released.

by Bill Clancy

Better than Basic

Can you program in a computer language other than Basic?

Enter this challenging new competition and win a Jupiter Ace.

Basic, for all its advantages, is slow. Programs written in Basic tend to look rather pedestrian when compared to programs written in some other languages such as machine code.

We want something different, something faster than Basic. It could be machine code, Forth, Lisp, Pascal or Fortran. In fact, your entry can be written in anything that is not Basic. And the best non-Basic program, be it game, utility or other, will win the Jupiter Ace.

The entries will be judged by *Popular Computing Weekly* editor Brendon Gore, and Jupiter Ace designers Richard Altwasser and Steve Vickers. In their selection account will be taken both of the standard of the program and of the accompanying documentation. The whole range of languages and types of program are allowed. The only stipulation is that it must not be written in Basic.

Entries to the award scheme must be accompanied by four of the numbered coupons published in *Popular Computing Weekly* throughout October. The closing date for the competition is November 18. The winning entry will be announced in the issue published on December 23.

Rules

1. *There is no limit on the number of entries you can send in, but each entry must be accompanied by four differently numbered competition coupons.*
2. *Closing date for entries is November 18, 1982.*
3. *The names of the winners will be announced in the December 23 issue of Popular Computing Weekly.*
4. *The Judges' decision is final.*
5. *No employees of Sunshine Publications Ltd or their families will be eligible to enter the competition.*

Popular Computing Weekly
Better than Basic Competition

Fill in this coupon. When you have collected four differently numbered coupons, send them with your program to: *Popular Computing Weekly* Better than Basic, Hobhouse Court, 19 Whitcomb Street, London WC2

NAME

ADDRESS

1

Spectrum

In this slot various contributors explore different aspects of the ZX Spectrum

Plotting the implications of a fast draw

***Malcolm Davison** explains how you can draw ellipses without slowing down*

If you like writing your own programs, and particularly if they are games programs, the chances are that at some point you will want to draw an ellipse. If you have tried plotting the formula for an ellipse which is

$$\frac{y^2}{n^2} = 1 - \frac{x^2}{m^2}$$

(where m and n are half the length of the major and minor axes, respectively) you may have found that it is not quite as straightforward as you would have wished.

If you apply equally stepped values of x to evaluate y the ellipse will not be complete unless the increments are very small and the plotting as a result painfully slow. A more satisfactory solution is to first evaluate y using increments of x and then evaluate x using increments of y or

$$y = \sqrt{n^2 - \frac{n^2x^2}{m^2}} \text{ and } x = \sqrt{m^2 - \frac{m^2y^2}{n^2}}$$

Now if you try the 'ellipse' program you will find the quadrants are plotted separately (see the diagram). The points where the curves meet are determined by setting the *For ... Next* loops (lines 90 and 130) to values so that there is neither overlap — nor a gap in the curve.

When it comes to producing a solid ellipse, simply drawing along the x or y axes between pairs of points plotted on the ellipse, even if done for both x and y axes will still produce a hole in the middle. So in ellipse 2 I have filled the central area as a separate operation, and at the point where the formula has been switched from values of x to y.

Notice how convenient the *Define Function* feature is when lengthy formulae are used. The use of *Gosub* abd *Return* also help keep the mainline routines simple and allow the subroutines to be re-used elsewhere in your programs.

So now we have drawn a good ellipse but because of the enormous amount of calculation and the lethargic pace of the Spectrum, it takes 30 or more seconds to complete. This is where ellipse 3 comes in — this first draws an ellipse, but then looks within the general confines of the ellipse and line by line notes the location of the circumference of the ellipse in a number array. The program then clears the screen and shows how quickly an ellipse can be drawn when the calculations have already been done (lines 700 — 730).

The program searches for the ellipse in lines 600-655, using the *Point* statement. Actually it only looks for the top left corner of the ellipse and reconstructs it, assuming symmetry on both axes. Just to show off, I have added lines 750 to 760 so that you can save the array on tape and verify it.

Once you have the array stored on tape which you can then feed into your program, remember to start the program with a *Goto* statement rather than *Run* (which will cheerfully clear the array you have just loaded). When you *Save* the program the arrays will be stored with it.

Suppose you want to send your program to *Popular Computing Weekly*? Well, the best way to pass this data on in listing form would be to store it in a *Data* statement and access it in the usual way using *Read*. To help you do this, lines 800 and 801 list the values in the array.


```
   1 REM ellipse
  60 DEF FN v(n,a,s)=INT SQR ((a
↑2)-((a↑2/n↑2)*(s↑2)))
  70 DEF FN x(n,a,s)=INT SQR ((n
↑2)-((n↑2/a↑2)*(s↑2)))
  80 LET a=32  LET n=24  LET r=8
8  LET s=87
  90 FOR s=0 TO 30 STEP 1
 100 LET y=FN v(a,n,s)
 110 GO SUB 400
 130 FOR s=24 TO 0 STEP -1
 140 LET x=FN x(a,n,s)  GO SUB 5
00
 150 STOP
 400 PLOT r-s,s-y  PLOT s+r,s+y
 410 PLOT r-s,s+y  PLOT s+r,s-y
 420 NEXT s  RETURN
 500 PLOT r-x,s-s  PLOT r+x,s+s
 510 PLOT r-x,s+s  PLOT r+x,s-s
 520 NEXT s  RETURN
```

```
   1 REM ellipse2
   2 PAPER 2  INK 3  BORDER 2  C
LS
  55 DEF FN v(n,a,z)=INT SQR ((a
↑2)-((a↑2/n↑2)*(z↑2)))
  70 DEF FN x(n,a,z)=INT SQR ((n
↑2)-((n↑2/a↑2)*(z↑2)))
  85 LET a=76  LET n=85  LET r=1
27  LET s=87
  90 FOR s=0 TO 86 STEP 1
 100 LET y=FN v(a,n,s)
 110 GO SUB 405
 200 FOR z=1 TO 70
 210 PLOT r-s,s+y-z  DRAW 2*s,0
 220 NEXT z
 240 FOR s=83 TO 0 STEP -1
 250 LET x=FN x(s,n,s)  GO SUB 5
00
 350 STOP
 400 PLOT r-s,s-y  DRAW 2*s,0
 410 PLOT r-s,s+y  DRAW 2*s,0
 420 NEXT s  RETURN
 500 PLOT r+x,s-s  DRAW 0,2*s
 510 PLOT r-x,s-s  DRAW 0,2*s
 520 NEXT s  RETURN
```

```
   1 REM ellipse3
   2 PAPER 0  INK 7  BORDER 2  C
LS
  50 REM
  51 REM **********************
  52 REM   Use ellipse function
  53 REM   to draw an ellipse
  54 REM **********************
  55 REM
  60 DEF FN v(n,a,s)=INT SQR ((a
↑2)-((a↑2/n↑2)*(s↑2)))
  70 DEF FN x(n,a,s)=INT SQR ((n
↑2)-((n↑2/a↑2)*(s↑2)))
  80 LET a=32  LET n=24  LET r=1
27  LET s=87
  90 FOR s=0 TO 30 STEP 1
 100 LET y=FN v(a,n,s)
 110 GO SUB 400
 200 FOR z=1 TO 36
 210 PLOT r-s,s+y-z  DRAW 2*s,0
 220 NEXT z
 240 FOR s=23 TO 0 STEP -1
 250 LET x=FN x(a,n,s)  GO SUB 5
00
 350 GO TO 600
 400 PLOT r-s,s-y  DRAW 2*s,0
 410 PLOT r-s,s+y  DRAW 2*s,0
 420 NEXT s  RETURN
 500 PLOT r+x,s-s  DRAW 0,2*s
 510 PLOT r-x,s-s  DRAW 0,2*s
 520 NEXT s  RETURN
 590 REM
 591 REM **********************
 592 REM   Store x co-ordinates
 593 REM     in the z() array
 594 REM **********************
 595 REM
 600 DIM z(87)
 610 FOR b=1 TO n
 620 FOR a=(r-m) TO r
 630 IF POINT (a,s+n-b)=1 THEN G
O TO 650
 640 NEXT a
 650 LET z(b)=a
 655 NEXT b
 656 CLS
 690 REM
 691 REM **********************
 692 REM   Draw the ellipse
 693 REM   using the array
 694 REM **********************
 700 FOR s=1 TO n
 710 PLOT z(s),(n+s)-s  DRAW (r-
z(s))*2,0
 720 PLOT z(s),(s-n+s)  DRAW (r-
z(s))*2,0
 730 NEXT s
 731 REM
 732 REM **********************
 733 REM  Save the array on tape
 734 REM **********************
 735 REM
 750 SAVE "ellipse" DATA z()
 755 PRINT "save now complete-re
wind tape - press play to verify
"
 760 VERIFY "ellipse" DATA z()
 800 FOR s=1 TO n  PRINT z(s)
 801 NEXT s
```

Programming

Boxing clever for the court in 30 bytes

John Durst presents a program for drawing boxes on your ZX81

Many games played on a computer take place on some kind of "court" many programs can be enhanced if results are displayed on a nicely divided format. Writing these "boxes" into a program in the normal way can use up a lot of memory To hold a full screen in a program uses nearly 700 bytes — a lot to sacrifice even when you do have the full 16K

But here is a program in machine code which will let you draw pretty well any box shape you want in just 49 bytes You can easily get it into 1K You will have to supply it with data — but even quite a complicated box will only use another 20 or 30 bytes

The idea is quite straightforward but you have to know and understand the rules I will try and make them as simple and clear as possible

First the operating program itself This is placed in a Rem statement in line 1 of your program (I am assuming that you have at least a rough idea of machine code programming) Fig 1 shows how to enter the machine code The actual program is in line 10 in hexadecimal Line 1 is set up with exactly 49 figures (the actual figures do not matter but I find it convenient to enter them a line at a time — that way I know I've got exactly 32 bytes for each digit)

If you enter the program exactly as shown in Fig 1 and then *Run* it you should get Fig 2 Line 1 will change into gobbledegook, but do not worry — it is still only using 49 bytes

Now you can scrap line 10, which has done its job, and get it ready for a new lot of data. You should also change the address in line 40 as you want to *Poke* the data to line 2 of your program Finally you must set up a new Rem statement in line 2 as a home for the data we have talked about I suggest a full line of 1 s (32 in all)

Before I explain how to code the data, why not type in line 10 in Fig 3 just to convince yourself that it works Type in lines 100 and 110 too — and then *Run* 10

Now the mystery formula for entering the data The code works like this A byte is expressed in hex notation as two hex digits (0 through F) The code uses the first hex digit to show how many times you want a character repeated ie if you want a line of 10 characters you would enter it as "A The second hex digit gives the character to be printed in that line as shown in Fig 4 Normal line

So if you want to rule a line 10 characters long you enter "A3 (you'll find that in positions 5 and 6 in line 10 of Fig 3) A0 gives 10 blanks and 17 gives a right-handed corner — you will find them at the start of line 10

Now comes the clever bit When you want to end a line you use the hex digits from 8 to F, in the second position of the byte This will give an inverse video character as shown in Fig 4 and signals the program to start a new line

The first digit you enter in this byte gets a different job, as well Instead of determining how many times the character is repeated it signals how many times the next line is to be repeated In line 10 of the program you will find "3C" — character C (the left-handed corner) for the end of the line and "3" to give 3 blank lines with vertical edges (the upper part of the box) This line is coded as "A0 15 A0 1D

The "1", in "1D", means the next line (the line of dots) will only appear once If it were "2D, it would be printed twice You probably realise that the opening "A0 in each line is there to displace the figure into the centre of the screen

One final thing the last two digits of data must be '00 This tells the program to stop Also the top line of any box (the first line of data) will only be printed once This is set up in the program and you cannot change it easily

Once you get the hang of it you will find this little program can be very versatile Fig 5b is derived from Fig 5a by changing just a couple of values I reckon you could set out a complete tennis court in 36 bytes of data

This program is particularly pleasing for computer buffs who are really into machine code It gives them the chance to use instruction RRD (ED67) — rotate right decimal — which must be the fanciest in the instruction set but hard to find a use for

Fig 1

```
   1 REM 11111111111111111111111111
11111111222222222222222222
  10 LET A$="8E01118940E1214800E1
A773E008E2002D1C9ED67C85F200748D
718FD1318EAEE880073E788BD78D28030
118DC4E13E118D04
  20 FOR J=1 TO LEN A$/2
  30 LET X=J*2
  40 POKE 16513+J,CODE A$(X-1)*1
6+CODE A$(X)-476
  50 NEXT J
```

Fig 2

```
   1 REM * RNDSSANDSTR$ ,?/
*SGN TAN GOSUB ?ACS ?4 ?NOT ( CL
EAR / REM INPUT NOT Y
 216 SGN /)=?( LPRINT /USR
  10 LET A$="8E01118940E1214800E1
A773E008E2002D1C9ED67C85F2007450
718FD1318EAEE88D73E788807800260030
118DC4E13E118D4
  20 FOR J=1 TO LEN A$/2
  30 LET X=J*2
  40 POKE 16513+J,CODE A$(X-1)*1
6+CODE A$(X)-476
  50 NEXT J
```

Fig 3

```
   1 REM * RNDSSANDSTR$ ,?/
*SGN TAN GOSUB ?ACS ?4 ?NOT ( CL
EAR / REM INPUT NOT Y
 219 SGN /)=?( LPRINT /USR
   2 REM 11111111111111111111111111
1'1111111
  10 LET A$="A017A33CA01EA01DA01
5A13CA015A01DA0A32300"
  20 FOR J=1 TO LEN A$/2
  30 LET X=J*2
  40 POKE 16508+J,CODE A$(X-1)*1
6+CODE A$(X)-476
  50 NEXT J
 100 PRINT AT 7,0;
 110 RAND USR 16514
```

Fig 4

CODING FOR BOX

NORMAL LINE	END LINE
0	8
1	9
2	A
	B
4	C
	D
5	E
	F

Fig 5a

Fig 5b

Machine Code

Ian Stewart and Robin Jones present a new series for beginners

Mneme and the micros

What can be held in a memory word? Well, any pattern of 16 bits, but those 16 bits can mean anything we want them to mean. If we want them to mean a 2's complement coded integer, then a word holds a number in the range — 32768 to 32767. If we want them to mean a positive integer with no sign bit then the number is in the range 0 to 65535. If we want, we can split the word into two 8-bit fields, each of which represents an alphabetic, punctuation or graphics symbol. As Tweedledee (or was it Tweedledum?) said 'When *I* use a word, it means just what I choose it to mean — neither more nor less.' Perhaps Lewis Carroll was ahead of his time.

Now for the special-purpose A-register. This is used every time you do any arithmetic. The result of any sum you ask the machine to do is put into the A-register (sometimes it's called the *accumulator*). Most arithmetic operations work on two values — it's no good asking the machine to work out 3+; you need to say what 3 is to be added to. One of these values must be in the A-register before the addition operation is executed. So you can write an instruction such as

```
ADD (1A3)
```

and the machine takes that to mean

1) Add the contents of memory location 1A3 to the contents of the A-register (the brackets round 1A3 indicate that it is the *contents* of 1A3 and not the *number* 01A3 which is to be added).
2) Put the result back in the A-register.

We have just written our first machine level instruction. It's not actually in machine code, but it's close. Look at its general form. It consists of an operation code *Add* and an address (1A3). Many instructions will look like that.

Incidentally, life is too short to say "operation code" too often. Everybody shortens it to *opcode*.

An addition program

Let's think about a sequence of machine instructions which would model the Basic statement

```
LET R = B + C
```

First we would have to assign actual addresses to *R*, *B* and *C*. Suppose that these are 103, 104 and 105 respectively. We have to get the contents of 104 into the A-register. Let's invent an *LD* (load accumulator) instruction to do this

```
LD (104)
```

and add the contents of 105

```
ADD (105)
```

and finally we need a way of storing the A-register's contents back in 103. So we'll invent a 'store' instruction

```
ST (103)
```

Now we have a simple machine level program consisting of 3 instructions

```
LD (104)        [load B into A register]
ADD (105)       [add on C]
ST (103)        [put the result in R]
```

How do we get the machine to run such a program? We are used to the idea that a program is stored in the machine *before* it's executed. After all, if you wrote the Basic statement

```
10 PRINT "HELLO WORLD"
```

you'd be somewhat disconcerted if, as soon as you hit *Newline*, the message "HELLO WORLD" were displayed. You expect it to be held until you need it. So by the same token, a machine level program has to be stored first. Where more natural to store an instruction than in a memory word (a word means what you want it to mean — remember)? Of course, that implies that the opcodes *Ld*, *Add* and so on have to be coded as bit patterns, but all we have to do is invent a table of bit patterns in a quite arbitrary way like this

Opcode mnemonic	Binary code
ADD	0000
LD	0001
St	0010

and every time we think of a new opcode that's needed, we add it to the table.

This assumes that all opcodes have a 4-bit binary code. That allows 16 different patterns and therefore 16 distinct instructions. This is a small instruction set by modern standards but it will do for our hypothetical toy computer. We've got 16 bits in the word altogether, so 12 are left for the address portion of the instruction.

So *Ld* (104), once inside the machine looks like

```
0001        000100000100
opcode      address (104 hex converted to binary)
```

Once you've seen one bit pattern, you've seen them all. From now on we'll write the hex versions of instructions. It's marginally less tedious.

Suppose we store our 3-instruction program from location 0FF onwards.

Contents	Address
	0FE
1104	0FF
0105	100
2103	101
	102
	103
	104
	105
	106

Now we need a way of saying to the machine 'Kick things off by executing the instruction in 0FF, then do the one in 100, then one in 101.' That's what the PC register or *program counter* is for. It acts as a kind of bookmark for the computer. We run the program in initialising the PC to the address of the first instruction. While the machine is obeying this instruction, the PC is automatically updated by 1, so that when the system returns to examine the PC, it will go and obey the next instruction, and so on.

There's a snag, though. While the last instruction (in 101) is being dealt with, the PC will be updated by 1 as usual, and so when the machine looks at it again, it will find 102, and leap off to execute the instruction there. What instruction? We didn't put one in 102. Ah! But there has to be a bit pattern in 102 left by a previous program, or just set up when the machine was switched on. So the machine will interpret this pattern as if it is an instruction, because that's what we've asked it to do. And then it will roll on through locations 103, 104 and 105 and that's where we're storing data. So if the number in 104 is 20FF, for instance, the machine will interpret this as

```
ST (0FF)
```

which will copy the contents of the A-register into 0FF, thereby destroying the first instruction of our program! Obviously what we need is a 'halt' instruction (we will use the mnemonic *HLT*) which stops the updating of the PC in its tracks. So the program now reads

```
LD (104)
ADD (105)
ST (103)
HLT
```

There's an important point to remember here. Precisely because we are using words to mean different things at different times, we have to keep a very careful eye on the implications the machine will draw from what we tell it to do. If we request it to *ADD* the contents of a location to the A-register, then it will assume that that location holds a number. It will make no tests, it cannot — any bit-pattern could represent a number. Similarly, any bit-pattern could represent an instruction, so if the PC points to a location, its contents will be executed as an instruction.

The rule is: *keep data and programs firmly apart*. If you don't, you can expect to be totally mystified at regular intervals. A whole program can disappear without trace while it is running.

To be continued next week

Reproduced from *Machine Code and better Basic*, by Ian Stewart and Robin Jones (price £7.50), by kind permission of Shiva Publishing Ltd, 4 Church Lane, Nantwich, Cheshire CW5 5RQ.

If you have any machine code subroutines/tips/games, please send them to Machine Code, *Popular Computing Weekly*, Hobhouse Court, 19 Whitcomb Street, London WC2 7HF.

Peek & poke

Peek your problems to our address. Ian Beardsmore will poke back an answer.

BINATONE WE KNOW NOT

Barry Keating of Harlow Essex, writes

Q After reading your news item about the Binatone (*Popular Computing Weekly*, July 15), I decided to wait for it But, recently neither you nor anyone else has carried any information on the Binatone for some time Do you know when it is going to come out? Failing this, have you any details on the Oric 1? Is it worth waiting, or should I go ahead and order a computer that is already on the market, such as the new Commodore 64, or ZX Spectrum?

A My advice is to go ahead and look for a computer that is available now We have tried to follow up the Binatone story but we have no new information The only two new micros liable to appear in the next few weeks are the Jupiter Ace and the Lynx Both these machines were at the Barbican show, September 9-12

WAITING, WAITING

James Knights of Badger Bank Ipswich, Suffolk, writes

Q I have been waiting for my Spectrum for many months and I am now very frustrated I have been thinking about getting another computer and have been looking around for one I am quite interested in buying an Atari 400, and I wonder if you could answer a few questions about it for me

Does it have a *Verify* command, does it have a flashing ability and how many colours can it display at once? Also, the Spectrum leaflet says that you cannot define your own graphics Is this true?

A If you want a computer which is geared to playing games then the Atari is a computer to consider But keep in mind the cost of Atari cartridges

I would also check if you can get your money back from Sinclair You might find that after all this time it is going to be quicker to get your Spectrum rather than cancelling your order and reclaiming your money It does seem at last as if some of the large backlog of orders is being cleared

The Atari does not have a *Verify* command, nor a flashing ability It can only display five colours and does not have a user defined graphics function as such To go some way towards compensating for these defects, the Atari has a higher resolution than the Spectrum and much better sound Each of the colours has a luminance which is essentially a *Bright*/*Dim* level that has 15 different variations

User defined graphics are also possible either by using a *Poke* command or else by using a character set generator module (16K) with a joystick

NO VINTAGE PROBLEM

H Marsland of Broomfield Close Chelford Macclesfield, Cheshire writes

Q I have just bought a ZX81 and, though I have two degrees (1935 vintage), I am finding the manual difficult to understand I am now at the stage where I can *Save* a program onto tape, and then get it to go back on to the computer.

Today I bought *Popular Computing Weekly* and found it full of discussion on the ZX Spectrum I wondered if you had any further details, and whether or not you think it advisable to buy one.

A Do not worry about not understanding the ZX manual Many people who have degrees of far more recent vintage than yours have found difficulty with the manual The books I usually advise to help out are *Getting Acquainted with your ZX81* by Tim Hartnell, from Interface 44-46 Earls Court Road London W8 and *Byteing Deeper into your ZX81* by Mark Harrison from Sigma Press Alton Road Wilmslow Cheshire SK9

If you have read further issues of *Popular Computing Weekly* you will probably have gained an idea as to what this latest Sinclair computer can do As to whether or not you think that you should buy one, my advice is to wait until you have a good understanding of your ZX81 A Spectrum can do everything a ZX81 can do, and a lot more besides

LINKUP EARMIC

David Heath of Bagley Close, Kennington Oxford writes

Q I am wondering if it is possible to link two ZX81s together via the Ear and Mic sockets? Then programs could be played from one computer to another, without having to *Save* and *Load* cassettes I would be very grateful if you could give me any advice

A It is worth trying but I cannot tell you what would happen as I do not know of this being tried However, it could not be done directly because of the signal levels of the Mic and Ear sockets

I would suggest that you put the computer with the program into *Save* mode so the signal is going down the Mic lead The output of this socket is very low so a small amplifier would be needed to boost the signal up to the 4-6 volts required by the Ear socket

Next, put the computer that is receiving the program into *Load* mode This is the only way that I could see you succeeding If you try it, please let me know the results, whatever the outcome

While on this subject, *J R Patterson of Pembroke Avenue, Great Yarmouth*, wants to know if a ZX81 can be used with his Sony TC 630 reel to reel tape recorder And *Robert Fander of Castle Lea Newport, Gwent* wants to know about using the Spectrum with his Phillips N2210 which has Din standard plugs

In both cases it is a matter of checking the output of the sockets which must be 4-6 volts If the reel to reel runs directly off the mains and does not have a transformer then it will probably output at 240 volts which will not do your 4-6 volt computer input a lot of good

Din standard plugs on the other hand tend to work on a very low output usually less than two volts This would be insufficient and as mentioned earlier some sort of amplifier would be needed

I'VE FINISHED WAITING

Jane Kennedy of Newton Mearns Glasgow, writes

Q I have just got my BBC micro, after weeks and weeks of waiting, and I think that it will be worth the delay I have been told by someone that it is possible to specify your own modes on the BBC computer, is this correct? If so, can you give me a listing, or a book, that will explain it to me?

A It is possible to specify your own modes by altering register six of the 6845 chip This is the register that sets up the number of character rows in a frame It can have a value from 0 to 127, though there are some important limitations which can lead to parts of the line being displayed off the screen or lines being repeated The book that deals with this subject in detail is *The BBC Micro revealed* by Jeremy Ruston, available from Interface publications

ONE BITE OF AN APPLE

Simon Harriss of Greencroft, Brampton Cumbria, writes

Q I have had considerable experience on the Apple II micro I have recently bought a BBC micro and would like to think that I have got to grips with it One thing still puzzles me. Can you tell me, simply, how to save and record data on cassette file? I cannot make sense of the provisional user guide instructions

A By now you should have your guide I have not seen it yet but I hope it is clearer than the provisional guide

You open a file with

```
A=OPEN OUT ( name )
```

where A is the variable and *Open out* can be replaced with *Open in* This is then followed by

```
PRINT #A data
```

or

```
INPUT #A, data
```

The file must be closed with

```
CLOSE #8
```

Classified

BOOKS FOR DRAGON 32
AND TANDY COLOUR COMPUTER

101 Colour Computer Programming Tips and Tricks £8 95
Colour Computer Programmes Home School Other £6 95
More Programmes Home School Other £6 95
The Colour Computer Songbook 40 songs £5 95
My Software (Poem Generator) £8 95
Colour Computer Graphics £7 95
Other books for PC1211 PC1500 Casio FX702 and Apple
Please add 50p per book post and packing maximum £1 50
Sole UK agents Dealer enquiries welcome

ELKAN ELECTRONICS
Free post (No stamp required)
28 Bury New Road Prestwich
Manchester M25 8L2
Tel: 061 798 7613 Prestel No 1 ... 2022

SPECTRUM CHARACTER GENERATOR Modify selected characters or re-define the complete character set with an 8X magnification display Save your character sets on tape for later use If you need specialist alphabets or graphic characters for games then this program is a must Cassette with full instructions £3 45 J Taylor 28 Azalea Close London W7 3DA

ZX81 GAMES KEYPAD Cased unit connects simply to computer Enhances programs greatly £5.99 G Christy 9 Indells Hatfield Herts

SPECTRUM SOFTWARE Five Arcade type games, including Planet Base Alpha (using hi res) All original for £3 95 D Roberts 27 Rashbrook Avenue Shrewsbury

ZX81 16K plus software and workcentre £110 ono Oswestry (0691) 4835 evenings

VIC20 Lightpen £8 44 columns £5 + arcade quality software (M C) List Grabosch Rubensstr 10 Hanover Germany

ZX81 PROGRAMMERS exchange Basic Tricks and program Sae + 10p stamp Nick Godwin 4 Hurkur Crescent Eyemouth Berwickshire

ACORN ATOM 12K + 12K 4K ROM Toolkit tapes books £175 Also O U Hekky with units, £100 232 Kingsway Burnage Manchester 19

2K RAM ZX81 5 games £6 2K RAM upgrade £6 50 Mr T Wood 194 Hagden Lane Watford Herts

ADVENTURE FOR VIC20 + 3K on C60 cassette only £4 95 G Allick 8 Amesbury Crescent Middlesbrough TS8 9HR Tel (0642) 590045

SPECTRUM Munchman Monster Maze Othello Zombies Ski Dictator Mastermind £3 95 each A Shaw 8 Yeadon Drive Southowram Halifax W Yorks HX3 9QX

SPECTRUM 48K Fruit Machine nudge hold gamble facilities on cassette with Roadrace only £1 50 A Haigh 70 Bradford Road Guiseley Leeds LS20 8LA

WANTED April issue of Laserbug Newsletter Write W Thompson 2 Sandown Close Sandy Beds SG19 1QT

NEW GENERATION SOFTWARE

from the authors of
3D MONSTER MAZE
and **3D DEFENDER**
for the ZX81 comes a game for the 16K

SPECTRUM

ESCAPE — Can you escape from the maze inhabited by five dinosaurs including a TRICERATOPS who hides behind the hedges and a PTERANODON who soars over the maze to swoop down on you?

Please order £4.95 to
NEW GENERATION SOFTWARE (Dept PCW)
FREEPOST (no stamp required)
OLDLAND COMMON BRISTOL BS15 6BR

BBC "TESS Are you bursting with imagination and look for graphic designing or just wanting a completely different type of program? TESS is for you. Design a shape choose colours pattern and size then watch the screen fill with copies of your design breathtaking! Model B only £9.50 inc. Cheque/PO Transcash no 61 413 1797 H&H Software Dept W 53 Holloway Runcorn Cheshire

ZX81 1K manuals PSU leads tape books £25 0284 64301

ZX81 16K Sinclair built, including manual £60 Tel Wilmslow (0625) 525668

TRS.80 L11 16K PSU, green screen monitor £200 of software only £175 0284 64301

SPECTRUM PROGRAMMES CASSETTE Assassin Maze Tanks Hunted Battleships Breakout Graphics Arithmetic £3 25 Cheque or PO to A Hipgach 43 Folgate Road Heacham Norfolk

MEMOTECH 64K MEMOPAK — brand new unopened and unused — unwanted prize just arrived £85 immediate delivery David Stevenson Fairmile Brookfield Crescent Stranraer DG9 0HY

VIC20 plus 3K Ram Joystick plus £30 worth of programs including Packman and Invader Fall plus VIC Revealed, worth over £270 will accept £150 ono Tel 061 225 9315 6—7 pm

ZX SPECTRUM AND SHARP MZ-80K SOFTWARE LIBRARY Low cost weekly hire from only 50p Send stamp for free catalogue Kernow Software 55 Eliot Drive St Germans, Saltash Cornwall

VIC20, cassette 3K Ram pack Joystick, software etc excellent condition £220 Hastings 440150 after 5 pm

ZX81 16K SOFTWARE five games machine code Meteor and Death Star (includes MC listwrite) Basic games Snake Maze and Dragon Maze only £4 69 from M Sudworth 17A Crowell Road Oxford

ZX81 VIDEO INVERTER Saves your eyes increases safe level displays sharp white characters on solid black background screen Kit £4 built £5 (includes VAT and P&P instructions) Reviewed in Popular Computing Weekly August 26 Send cheque postal order to D Finch 5 Stanton Road Thelwall Warrington Cheshire WA4 2HS

ZX80 8K ROM 16K RAM manuals, programs £60 Northwich 3908

SPECIAL OFFER!!

the new DRAGON 32

Get switched on to a 6809E colour cpu with hi res graphics and quality keyboard!
Just £195 inc VAT (+ delivery £2)
OR
Send your working ZX81 and £175 to us instead before Clive cuts the price again !?

H COOMBS
4 Northgate Drive, Camberley Surrey GU15 2AP

BBC SOFTWARE
Educational and Leisure programs
Space Academy 32K Driving Test 32K Golfmania 32K England 32K Battle Ships 32K Film Buffs 32K and more Programs £4 inc 2 for £6 inc S A E for details
Sent by return of post after cheque/POs cleared Mail order only
SWIFT LINK SOFTWARE
118-120 WARDOUR STREET W1V 4BT

STACK STOREBOARD for Vic, 3K+16K Ram Cost £89 sell for £65 ono Tel 0740 53133

FOR PC-1211 Six listings including Calendar for any year (need printer) only £1 to R Haydon 21 Manchester Mansions London N19 3SW

ZX81 VENTURE Genuine 16K adventure type game Fantastic screen effects on cassette £5 Bolaker 29 Chadderton Drive Unsworth Bury Lancs

BBC MICRO Pools Predictor Model A or B Easy data entry Uses powerful mathematical and statistical forecasting model User turntable facility On cassette with full instructions £4 99 Mayday Software 181 Portland Crescent Stanmore HA7 1LR

CHEMISTRY, 4 different programs covering formulae moles calculations volumes (for **ZX81** or Spectrum) Programs £5 each with extensive notes or send SAE for details

16K SPECTRUM, Air-Raid Simon Lunar-Lander All for £1 Send SAE plus rewound cassette to R S Allen Upway Corner Upway Chalfont St Peter Bucks SL9 0AG

DAI used demonstration only plus joystick Adventure cassette and handbook. Bargain £590 Everyman Computers Edward Street Westbury Wilts Tel 0373-864644

VIC20 COMPUTER Datassette Arfen expansion unit 16K RAM 3K RAM super expander and programmer's aid cartridges 5 games cassettes Several books Only £395 Telephone (0732) 850513

SPECTRUM OZ CONNECTOR No more plug pulling load save inbuilt mic speaker jackplug to amplify beep £10 see for details J Incledon Long Beach, Warren Road, Brean, Somerset. Tel 027-875477

ZX81 1K 12 superb games including Breakout and Invaders Original games Original graphics No quibble money back guarantee £5 On quality cassette Living Software, 3 Woodstock Close Wordsley Stourbridge West Midlands DY8 5HZ

UNEX VIC20 SOFTWARE. Pacman Space Snoot-out Space Raiders Missile Defence all in hi res graphs All on one cassette £5 50 0706 65 3325

WANTED DRAGON SOFTWARE

We will buy outright or pay a good royalty for any first class Dragon 32 Software
Phone (0279) 52401
Or write to

GEM SOFTWARE
22 PRESTWICK DRIVE
BISHOPS STORTFORD
HERTS

BYTE-MAN with Star fighter and Bomber — three quality fast moving machine code games for the price of one for the ZX81 16K, why pay more for less? Only £2 95 Mindseye 12 North Grove Drive Leeds LS8 2NJ

Computer Swap

SPECTRUM 48K, unused in original packing £175 Call Justin 01-499 0880 9am to 5 pm

GENIE 1 16K, as new Software + books £160 ono Tel Bristol (0272) 276853

VIC20 3.5K unwanted present only six months old In full working order still under guarantee £140 061 789 4120

FULLER 16K ZX81 + all leads and manual Worth over £140 Mint condition Will accept £60 Tel 051 536 3327 for details

UK 101 cased 8K RAM monitor 2 soundboard 300/600 Baud £130 Tel 01-455 7544

COMMODORE PET, 2001 series new ROM 32K tool-kit, superchip green screen. £400 ono Mr Newman 01 953 7320 (day or evening)

SPECTRUM 48K brand new best offer for immediate delivery Crowthorne (03446) 3475 (day or evenings)

VIC20 CASSETTE DECK 8K RAM cartridge books etc. Cost £320 Sell for £200 D Yardley 46 Park Road Chandlers Ford, Hants Tel Chandlers Ford (65) 4632

VIC20 COLOUR COMPUTER 20K RAM hi-res graphics ROM RS232 interface Cassette player joystick manuals and software £250 Kidderminster 515285

ZX81 16K, £65 Mr Chiv 59 Shaw Road Heaton Moor Stockport. Tel Stockport 431 0075 after 5 pm

SHARP MZ 80K. 48K + 2 basic extensions £250 Buyer to collect Ring Malcolm on Barnsley 43788 after 5 pm

ACORN ATOM 12K RAM 8K ROM several Bug Bye games. £130 or swap for Vic20 Mr Kimmis 50 Leamington Road Ainsdale Southport, Merseyside 0704 78533

VIC20 with cassette unit and 3K RAM £180 Telephone 0264 88283 before 5 pm.

ZX81 16K RAM 7 months old £65 including £35 worth of software Andrew Norris 10 Shee road London W3 Tel 743 6968

ZX81 16K In/outport, keyboard software 8 books bargain at £85 ono Mr D Jewitt, 28 Earls Court Gardens London SW5 01 373 2780

TI/59 WITH PRINTER — swap for Vic20 or TI/99 or similar micro Tel Mr Thiru 01 580 0055 ext 58 (daytime)

VIC20 WITH CASSETTE UNIT 3K + 16K RAM cartridges super expander cartridge monitor prog art + software flags books Worth £520 Yours for under £400 Mr De Rosa 01-882 0083

VIC20 SYSTEM comprised with tape unit and disk drive 20K RAM expansion plus games cartridges and cassettes also joystick 3 months old £500 ono Tel Romford 24557

CASIO FX602P Boxed almost new with manuals Offers invited Phone Hull (0482) 781517 after 5 30 pm (not Sundays Tuesdays Wednesdays)

ZX81 PLUS 16K RAM and £150 worth of software Yours for only £150 Phone Neil Caldwell on 061 773 5242 after 6 pm

ZX80 PLUS 8K ROM slow (Comp Shop simulator) + 16K RAM for £75 Phone Tony Hennor on 0482 834433

ATARI TV GAME SYSTEM with Combat and Space Invader cartridges rarely used Very good condition £80 ono Tel Medway (0634) 70145

VIDEO GENIE 16K RAM with selection of software £180 Contact G Turner 8 Vale Avenue Bury Lancs Tel 061-764 8417

16K ZX81 plus programs 4 months old £90 00 F Ford 8 Shelley Crescent Southall Middlesex.

ZX81 SINCLAIR 16K + manuals and lead also book and mags, £80 Tel Codsall 5691 Simon Darkside 12 Bromley Gardens Codsall Nr Wolverhampton Staffs WV8 1BE.

ZX81, 16K RAM pack Haven keyboard with keyboard repeat and sounder Inverse Inside plus case Yours for £130 Mr P Deuel 38 Timber Croft Lane London SE18 Tel 01-854 5854

UK 101 32 × 48, New basics 1345 Segman tool kit sound board cased 7 amp power supply PSU originally £300+ offers around £100 Mr Barker 0773 853258

DAI PERSONAL COMPUTER 48K six months old cost £684 Bargain at £390 Phone 0295 4847

VIC20 UNEXPANDED Perfect condition £200 ono G Williamson 12 Trafford Road Willerby Hull Tel (Hull) 650825

ZX81 SINCLAIR BUILT, unused fully guaranteed boxed with leads and manual and 16K, £50 ono Preferably collected Phone 01 566 1387 anytime

PET, SERIES 3001 8K external cassette deck plus many programs, £220 Tel 01 940 2077 after 4 pm

WANTED 16K RAM PACK, around £15 Tel Battle (04246) 2717

TRS-80 Level 1 4K Ram £110 ono G Carrett Newcastle-upon Tyne (0632) 852009

VIC20, cassette deck basic learner I £180 S Robinson 01 738 3860 (evenings)

UK101 8K cased 16K Rom assembler in Eprom basic 5 and sound £100 ono Tel 01 941 0730

POPULAR Computing WEEKLY

BACK NUMBERS

MAKE SURE OF A REAL COLLECTORS' ITEM — THE FULL SET OF *PCW*

We will mail any of the numbers you're missing from Issue 1 to the latest — for just **50p** an issue, including p & p

(We have no more copies of Issues 2, 6, 7 or 11)

Send cheques/Postal Orders to:

Back Numbers
Popular Computing Weekly
Hobhouse Court
19 Whitcomb Street
London
WC2 7HF

COMPUTER SWAP
01-930 3266

Do you want to buy or sell a microcomputer? You can do it FREE in Computer Swap, a new regular service for *Popular Computing Weekly* readers

All you have to do is phone Computer Swap on 01-930 3266 and give us details of your computer, the price you want for it, your name, address and telephone number

Computer Swap entries are limited to a maximum of 30 words They will be published in the first available issue

POPULAR Computing WEEKLY

CLASSIFIED ADVERTISING

Computer Swap — Free/Private reader — 10p a word/Trade Advertisement — 20p a word/Semi-display — £5 a single column centimetre, minimum two-column centimetres

Computer Swap — buy or sell your computer for free through Computer Swap See box on left for details

Private readers — other advertisements from private readers cost 10p a word

Trade advertisements — cost 20p a word

Semi-display — why not make your advertisement more substantial by choosing the semi-display rate It is only £5 a single column centimetre

Send your classified entries to Classified Department *Popular Computing Weekly*, Hobhouse Court, 19 Whitcomb Street, London WC2 For semi-display enquiries call Alastair Macintosh on 01-930 3840

Here's my classified ad.

Please continue on a separate sheet of paper

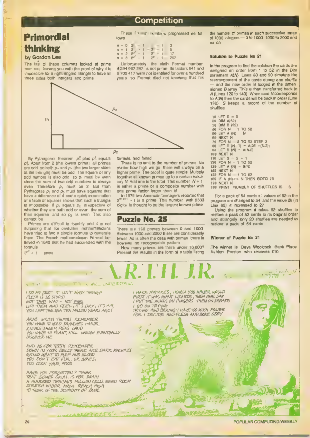
I make this ______ words at ______ per word so I owe you £ ______

Name ______

Address ______

______

Telephone ______



Primordial thinking

by Gordon Lee

The last of these columns looked at prime numbers leaving you with the proof of why it is impossible for a right angled triangle to have all three sides both integers and prime

By Pythagoras theorem p_1^2 plus p_2^2 equals p_3^2. Apart from 2 (the lowest prime) all primes are odd so both p_2 and p_3 (the two larger sides of the triangle) must be odd. The square of any odd number is also odd so p_1 must be even since the sum of two odd numbers is always even. Therefore p_1 must be 2. But from Pythagoras p_2 and p_3 must have squares that have a difference of 4 and a quick examination of a table of squares shows that such a triangle is impossible. If p_1 equals p_2 irrespective of whether they are both odd or even the sum of their squares and so p_3 is even. This also cannot be.

Primes are difficult to identify and it is not surprising that for centuries mathematicians have tried to find a simple formula to generate them. The French mathematician Fermat believed in 1640 that he had succeeded with the formula

$2^{2^n} + 1$ = prime

These Fermat numbers progressed as follows

$n = 0 \quad 2^{2^0} + 1 = 2^1 + 1 = 3$
$n = 1 \quad 2^{2^1} + 1 = 2^2 + 1 = 5$
$n = 2 \quad 2^{2^2} + 1 = 2^4 + 1 = 17$
$n = 3 \quad 2^{2^3} + 1 = 2^8 + 1 = 257$

Unfortunately the sixth Fermat number 4 294 967 297 is not prime. Its factors 641 and 6 700 417 were not identified for over a hundred years so Fermat died not knowing that his formula had failed.

There is no limit to the number of primes. No matter how high we go there will always be a higher prime. The proof is quite simple. Multiply together all known primes up to a certain value say N. Add one to the total. This number $N + 1$ is either a prime or a composite number with one prime factor larger than N.

In 1979 two American teenagers reported that $2^{21701} - 1$ is a prime. This number with 6533 digits is thought to be the largest known prime.

Puzzle No. 25

There are 168 primes between 0 and 1000. Between 1000 and 2000 there are considerably fewer. As is often the case with primes there is however no recognisable pattern.

How many primes are there under 10 000? Present the results in the form of a table listing the number of primes in each successive range of 1000 integers — 0 to 1000, 1000 to 2000 and so on.

Solution to Puzzle No 21

In the program to find the solution the cards are assigned an order from 1 to 52 in the Dim statement $A(N)$. Lines 80 and 90 simulate the rearrangement of the cards during one shuffle — and the new order is lodged in the dimensioned B array. This is then transferred back to A (Lines 120 to 140). When card N corresponds to $A(N)$ then the cards will be back in order (Line 160). S keeps a record of the number of shuffles.

```
10 LET S = 0
20 DIM A(52)
30 DIM B (52)
40 FOR N = 1 TO 52
50 LET A (N) = N
60 NEXT N
70 FOR N = 2 TO 52 STEP 2
80 LET B (N-1) = A(26 +(N/2))
90 LET B (N) = A(N/2)
100 NEXT N
110 LET S = S + 1
120 FOR N = 1 TO 52
130 LET A (N) = B(N)
140 NEXT N
150 FOR N = 1 TO 52
160 IF A (N)<> N THEN GOTO 70
170 NEXT N
180 PRINT "NUMBER OF SHUFFLES IS "; S
```

For a pack of 54 cards all values of 52 in the program are changed to 54 and the value 26 (in Line 80) is increased to 27.

Using the program it takes 52 shuffles to restore a pack of 52 cards to its original order and strangely only 20 shuffles are needed to restore a pack of 54 cards.

Winner of Puzzle No 21

The winner is Dave Woolcock, Bank Place, Ashton, Preston, who receives £10.